Order Number 9521304

Optical texture characterization

Phuvan, Sonlinh, Ph.D.

City University of New York, 1995

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

Optical Texture Characterization

by

SONLINH PHUVAN

A dissertation submitted to the Graduate Faculty in Engineering
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy, The City University of New York

1995

This manuscript has been read and accepted for the Graduate Faculty in Engineering in satisfaction of the dissertation requirement for the degree of the Doctor of Philosophy.

Nov. 28th 1994
Date

Chair of Examining Committee:
Professor Yao Li

11/29/94
Date

Co-Advisor: Professor Samir Ahmed

11/28
Date

Executive Officer: Dean G. Lowen

Professor Joe Barba

Professor Mitra Basu

Dr. Bruce Copeland

Supervisory Committee

THE CITY UNIVERSITY OF NEW YORK

Abstract

OPTICAL TEXTURE CHARACTERIZATION

by

Sonlinh Phuvan

Advisor: Professor Yao Li

A novel method, the polyfractal measure, for uniquely characterizing fractal texture is described. This technique presupposes that textures are produced by a linear combination of subtextures. Those subtextures are also fractals. There is an infinite set of subtexture combinations which can be used to describe the texture, but a set of subtextures is selected to provide for discriminating between textures. The set of the fractal measure of the discriminating subtextures is obtained, and this set can uniquely identify a texture. The discriminating subtextures are obtained by using a novel pattern classification technique, N-wavelet coding. A set of discriminating features which can be used to classify the textures are obtained using an artificial neural network. Those set of features are used to obtain a set of wavelet function which are used in the detection of those discriminating features (subtextures). The wavelet functions than can then be used to create the subtextures, from which the polyfractal measure can be obtained.

An optical technique is developed for implementing the characterization algorithms in real time, using optically addressed amorphous silicon ferroelectric liquid crystal spatial light modulators. All key subsystems, an optical binary to grey level processing and a real time optical iterative processor, are demonstrated.

LIST OF TABLES

LIST OF FIGURES

1 Introduction

In this dissertation a novel technique for uniquely characterizing textures is described and components to optically implement this technique in real time are demonstrated.

Applications of texture classification include; characterization of VLSI wafers for quality control of fabrication processes, material manufacturing quality and process control, terrain characterization for automated battle planning, automated identification of potential targets for cruise missile mission planning, identification and classification from multisensor data for mine warfare, and medical diagnostic applications such as mammograms, CAT scans, and MRI scans.

The approach taken in this thesis is to identify discriminating class features, and to obtain a scale invariant incidence measure of each features. The vector representing the incidence measure for each classification feature provides a unique characterization of a texture class. This is because a set of classification feature is selected in such a way that it is unique to a given texture class.

The identification of the discriminating class features, or the classification basis, is obtained by using a feedforward artificial neural network with backpropagation learning. An artificial neural network is essentially an associative memory, and it can be autoassociative or heteroassociative. In this dissertation, the neural network is set up as

an associative memory, it associates a particular input with an output class identification. The purpose of the artificial neural net in this dissertation is not to classify but to provide the feature basis. It is possible to obtain a feature basis without using a neural network. For example an orthogonalization procedure with one of the input as the basis. Thus for M classes on can obtain M! basis, if one of the M input classes is used as the first classification feature. If an arbitrary feature is selected then an infinite number of classification basis sets can be obtained. The question is not whether a classification basis can be obtained, but how good it is. The strenght of using an artificial neural net is that assuming that it is correctly set up it will provide a "sufficiently good" classification basis, without the need to make any apriori assumption as to the first classification feature, as is required in an orthogonalization procedure. An artificial neural net works by extracting the classification basis at the first neural layer where the input is initially presented to the artificial neural network. Each node in that first layer, produces an inner product of its weight and the input data, i.e. a zeroth displacement correlation between the input and a feature defined by the weight of that node. If the match is sufficiently good then the associated node of that classification feature becomes active indicating that this particular classification feature has been detected. Thus the first layer is a feature extraction layer. Those features are then subsequently used by the remaining layer for classification purpose. Since in order to successfully extract the classification feature the artificial neural net must converge, it might imply that the entire classification process can be executed using an artificial neural network. It is difficult to obtain scale invariant classification using an artificial neural netowork without apriori knowledge of the scaling

factor of the input. By using the classification features obtained by the neural network to construct a wavelet function classification basis, a scale invariant, position invariant, noise and distortion tolerant and sensitive classification method can be obtained. It is difficult to provide scale invariance in an artificial neural network, and to provide for noise and distortion tolerance a large training set comprising all possible instances of noisy and distorted inputs. Using a wavelet function, which is inherently noise and distortion tolerant, the need for extensive artificial neural network training is significantly reduced. The results obtained using artificial neural networks for feature extraction and using wavelet functions to do the feature detection combines the best parts of both and has interesting implications for the design of artificial neural networks.

The description of the texture characterization technique is limited to 1D textures, but it can be easily generalized to 2D textures. An interesting method is to use 1D features to classify 2D texture by mapping the 2D texture into 1D sequence. A scale invariant scanning technique is described which will allow 2D texture characterization using 1D feature vectors. This simplifies the design of a real time optical implementation of this texture characterization technique. At the time this research was performed, the state of the art in high speed optically addressed spatial light modulators were binary optically addressed ferroelectric liquid crystal devices. Implementation of the texture characterization algorithms require grey level processing. A method for obtaining grey level processing from binary optically addressed spatial light modulators is also described.

The dissertation consists of two key parts. The first key part, is chapter 3 on "Texture Classification". In this chapter the theories and their evaluations are described. This chapter is subdivided into 3 sections. Section 3.2 describes the "N-Wavelet Coding" algorithm to be used to identify and classify subtextures and/or patterns. Section 3.3 describes by using the N-Wavelet coding algorithm how it is possible to uniquely characterize fractal textures by producing a "Polyfractal" measure. Section 3.4 describes a method for characterizing 2D textures with 1D features using a fractal scanning method such as a Peano space filling curve. The second main part , Chapter 4, entitled "Optical Implementation", describes the subsystems used to optically implement the algorithms developed in the previous chapter. Section 4.2 describes how a binary spatial light modulator can be used to obtain the grey level processing required to implement the correlation operation to be used in a wavelet transform. Section 4.3 describes an optical iterative processor used to implement the iterative scaling of the wavelet function and the repetitive correlation operation to be used to implement the wavelet transform. This iterative processor can also be used to obtain an estimate of the fractal dimension.

2 Literature Survey

2.1 Introduction

In this thesis a texture is assumed to be defined by structures randomly embedded within a n-dimensional field. A texture is assumed to be a fractal, or have fractal components. In other words, a texture is self similar with respect to scale. The structures embedded within the texture can be considered to be fractal texture primitives. A fractal texture primitive does not exist in all texture classes.

This combination of deterministic (i.e., the texture primitives), and stochastic (i.e. produced by variance of scale, position and incidence) features in a texture makes it difficult to obtain a general method for classification. Several approaches have been proposed to classify textures, those techniques can be subdivided as follows: oriented texture fields[3,4,33,79] , phase portraits[29,65] ,syntactic[38,41] , symbolic[27,85] , and statistical[11,31,32,58,86].

The oriented texture field is essentially a contour map of the texture, where each contour is an isogradient, or line of constant gradient. Each point in the texture field is comprised of two elements, and angle image and a coherence image. The angle image indicate the dominant local orientation of a given point, and the coherence

image indicates the amplitude of the local maximum gradient and is a measure of the local anisotropy.

The phase portrait of a texture is obtained by assuming that the oriented texture field is the result of the phase plot of a differential equation. The texture field can be characterized by phase structures embedded within its field. The phase structures include limit cycles, critical points, saddle points, and spirals. A problem arises in that dissimilar texture class can have similar or identical embedded phase structures.

In the syntactic approach, the texture is classified by the language or words used to describe it. The grammar comprising the language is defined a priori. As yet no universal language has been developed which can characterize all textures.

The symbolic approach has similarities with the phase portrait and syntactic approach. In this approach, structures within the texture fields are not phase structures or syntax but symbols. Examples of this approach are in the fields of petrology, metallography and crystallography.

All the above technique requires order in the texture field, and break down for weakly ordered texture where texture primitives are not ordered, and are randomly placed within the texture field.

Several approach used to characterize weakly or randomly ordered textures are statistically based. Two such approach are the entropy and the fractal measure. The entropy measure is based on the cooccurence matrix. The element of this matrix is the second order joint probability density function that for a given intersample spacing, i.e. spacing between texture region, and direction of intersample spacing a given gray level transition occurs. The entropy measure is an estimate of the disorder of the texture. The fractal measure is a statistical measure of the perceived roughness of the texture. This measure is based on the relationship between detected elements and sampling size. There are many methods for estimating the fractal dimension of a texture, such as the box counting method[36], the power spectrum method[42], the difference statistics method[58], the covering blanket method[55], and the space filling scanning method[42,47,51].

The fractal measure has several properties which makes it a good candidate for characterizing textures, it is position and rotation invariant, and most importantly it is scale invariant. Thus in the following sections the fractal measure will be examined further.

2.2 Survey of Techniques for Estimating Fractal Dimensions

2.2.1 Fractal Dimensions of Textures

A complete rigorous definition of dimension was given in 1919 by Hausdorff[28], but is difficult to implement numerically.

Fractal measures (dimension) can be subdivided into three classes. First, dimension can be defined as a scaling of bulk with size. Second, fractal dimension can be thought as a coarse-grained volume. Third, the degree of freedom of strange attractors can be related to the information dimension.

2.2.1.1 Local Scaling Comparison of Bulk with Size

A geometrically intuitive notion of dimension is as an exponent that expresses the scaling of an object's bulk with its size:

$$\text{bulk} \sim \text{size}^{\text{dimension}} \tag{1}$$

The definition of dimension is usually cast as an equation of the form

$$\text{dimension} = \lim_{size \to 0} \frac{\log bulk}{\log size} \tag{2}$$

where the limit of small size is taken to ensure invariance over smooth coordinate changes. This small-size limit also implies that dimension is a local quantity and that any global definition of fractal dimension will require some kind of averaging.

The obvious relevant measure of bulk for a subset B of dynamical attractor is its natural invariant measure $\mu(B)$, although other notions of bulk are also possible and will be discussed. A good quantity for the size of a set is its radius or its diameter, the latter of which is defined by

$$\delta(B) = \sup\{|\mathbf{X}-\mathbf{Y}| : \mathbf{X},\mathbf{Y} \in B\} \tag{3}$$

where *sup* is the supremum, or maximum, and $|\mathbf{X}\text{-}\mathbf{Y}|$ is the distance between $\mathbf{X}$ and $\mathbf{Y}$. How this distance is calculated depends on the norm of the embedding space. If X_i is the i^{th} component of the vector $\mathbf{X} \in \mathbf{R}^m$, then the L_s gives distance according to

$$|\mathbf{X}-\mathbf{Y}| = \left(\sum_{i=1}^{m} |\mathbf{X}_i - \mathbf{Y}_i|^s\right)^{\frac{1}{s}} \tag{4}$$

The most useful of those norms are L^2, the Eucledian norm, which give distances that are rotation invariant; L^1, the taxicab norm, which is easy to compute; and L_∞, the maximum norm, which is also easy to compute. It is not difficult to show that the fractal dimension is invariant with respect to the type of norm.

2.2.1.1.1 Pointwise Dimension

The pointwise dimension is a local measure of the dimension of the fractal set at a point on the attractor. Let $B_r(x)$ denote the ball radius r centered at the point **X**. (Whether this ball is a hypersphere or a hypercube will depend on the norm L_s).

The pointwise mass function $B_r(x)$ is defined as the measure

$$B_x(r) = \mu[B_x(r)] \tag{5}$$

A measure μ is defined on the set $\mathcal{A}$ if the subsets $\mathcal{B}$ of the set $\mathcal{A}$ can be associated with real values $\mu(\mathcal{B})$ that represents how much of $\mathcal{A}$ is contained in $\mathcal{B}$. The measure that is defined on a set reflects the varying density over the set and can intuitively be regarded as mass[68].

The scaling of the mass function at function **X** with the radius r defines the pointwise dimension[18] at **X**

$$D_p = \int_A D_p(\mathbf{X}) d\mu(\mathbf{X}) \tag{6}$$

A useful measure μ should be invariant under time evolution:

$$\mu(\mathcal{B}) = \mu[\phi_{-t}(\mathcal{B})] \tag{7}$$

2.2.1.2 Coarse-Grained Volume and Hausdorff Dimension

The Hausdorff dimension is solely dependent on of the geometry of the fractal set and makes no reference to the a priori measure μ that may be defined on the attractor. The Hausdorff definition begins by defining a measure Γ, which corresponds to uniform density over the fractal set.

Let $\mathcal{A}$ be a fractal whose dimension is to determine. Let $C(r,t)=\{\mathcal{B}_1,\mathcal{B}_2,...,\mathcal{B}_k\}$ be a finite covering of $\mathcal{A}$ into sets whose diameters are less than r. That is, $\mathcal{A} \subset \bigcup_{i=1}^{k} \mathcal{B}_i$ and the diameter of each set satisfies

$$\delta_i \equiv \delta(B_i) < r \tag{8}$$

$$\Gamma(A, D, r) = \inf_{C(r,a)} \sum_i \delta_i^D \tag{9}$$

where inf is the infimum (or minimum) over all coverings satisfying $\delta_i \prec r$, defines a kind of coarse-grained measure for the set $\mathcal{A}$.

For example, if D=1 , then $\Gamma(\mathcal{A},D,r)$ gives the length of the set $\mathcal{A}$ as measured with a ruler of length r, and, as $r \rightarrow 0$, Γ approaches the actual length of $\mathcal{A}$. For most values of D, the $r \rightarrow 0$ limit leads to a degenerate measure: either $\Gamma \rightarrow 0$ or $\Gamma \rightarrow \infty$. That is not surprising: a figure with finite length will have zero area, and a finite area is covered by a curve of infinite length.

One can think of $\Gamma(\mathcal{A}, D)$ as the D-dimensional volume of the set $\mathcal{A}$. In fact, since $\Gamma(\mathcal{A}, D, r)$ is a function that decreases monotonically with D, there is a unique transition point D_H that defines the Hausdorff dimension:

$$\Gamma(A,D) = \lim_{r \to 0} \sup \Gamma(A,D,r) = \left\{ \begin{array}{l} \infty \text{ for } D < D_H \\ 0 \text{ for } D > D_H \end{array} \right\} \tag{10}$$

so that

$$D_H = \inf\{D : \Gamma(A,D) = 0\} \tag{11}$$

defines the Hausdorff dimension.

The coarse-grained measure $\Gamma(\mathcal{A}, D, r)$ typically exhibits the scaling $\Gamma(\mathcal{A},D,r) \backsim r^{D-D_H}$, which gives another way to estimate dimension.

Having defined the Hausdorff dimension D_H, we give the Hausdorff measure on the set $\mathcal{A}$ by $\Gamma(\mathcal{A}, D_H)$. The Hausdorff measure of a subset $\mathcal{B}$ of $\mathcal{A}$ is given by

$$\Gamma(B, D_H) = \lim_{r \to 0} \sup \inf C(r, B) \sum_i \delta_i^l \tag{12}$$

and this measure is one way to describe the bulk of the set $\mathcal{B}$.

This implies

$$\Gamma(B, D_H) \sim \delta(B)^{D_H} \tag{13}$$

which is similar to

$$\text{bulk} \sim \text{size}^{\text{dimension}} \tag{14}$$

This displays the role of the Hausdorff dimension as a local scaling exponent.

2.2.1.2.1 Box-Counting Dimension

Because of the infinum over all the coverings it is difficult to implement the computation the Hausdorff dimension. If this requirement is relaxed and one chooses a covering that is simply a fixed-size grid, one obtains an upper bound on the Hausdorff dimension, also known as the capacity, the box-counting dimension, or the fractal dimension.

For most fractal sets of interest, the capacity and the Hausdorff dimension are equal[18].

For grid size r

$$\Gamma(A, D, r) = \sum_i \delta_i^D = \sum_i r^D = n(r) r^D \tag{15}$$

where $n(r)$ is the number of nonempty grid boxes. The box counting dimension is the value of D on the transition between $\Gamma \rightarrow 0$ and $\Gamma \rightarrow \infty$.

$$\Gamma(\mathrm{A}, D_H, r) \sim 1 \tag{16}$$

implies

$$n(r) \sim r^{-D_H} \tag{17}$$

or more formally

$$D_H = \lim_{r \to 0} \frac{\log(1/n(r))}{\log(r)} \tag{18}$$

Here the local notion of bulk is replaced with a global one: $1/n(r)$ is the average bulk of each nonempty box, since each box contains, on average, $1/n(r)$ of the whole fractal.

2.2.1.2.2 Generalized Dimension

In computing the box-counting dimension, a box is counted if it contains at least one point. No provision is made for weighting the box count according to how many points are inside a box. The geometric structure of the fractal set is analyzed but the underlying measure is ignored.

Let $\mathcal{B}_i$ denote the i[th] box, and let Pi = $\mu(\mathcal{B}_i)/\mu(\mathcal{A})$ be the normalized measure of this box. Equivalently, P_i is the probability for a randomly chosen point on the attractor to be in $\mathcal{B}_i$, and it is usually estimated by counting the number of points that are the i^{th} box and dividing by the total number of points.

The generalized dimension is defined by

$$D_q = \frac{1}{q-1} \lim_{r \to 0} \frac{\log \Sigma_i P_i^q}{\log r} \tag{19}$$

Writing the sum of P_i^q as a weighted average

$$\sum_i P_i^q = \sum_i P_i \left(P_i^{q-1} \right) = \left\langle P_i^{q-1} \right\rangle \tag{20}$$

one can associate bulk with the generalized average probability per box $\left\langle P_i^{q-1} \right\rangle^{1/(q-1)}$ and identify D_q as the scaling of bulk with size. For q=2 the generalized average is the ordinary arithmetic average, and for q=3 it is a root mean square. It is not hard to show that the limit q $\rightarrow$ 1 leads to a geometric average. Finally, it is noted that q=0 corresponds to the plain box-counting dimension defined above.

For a uniform fractal, with all P_i equal, one obtains a generalized dimension D_q that does not vary with q. For a nonuniform fractal, however, the variation of D_q with q quantifies the nonuniformity. For instance,

$$D_{\infty} = \lim_{r \to 0} \frac{\log\left(\max_i P_i\right)}{\log r} \tag{21}$$

$$D_{-\infty} = \lim_{r \to 0} \frac{\log\left(\min_i P_i\right)}{\log r} \tag{22}$$

The notion of generalized dimension first arose out of a need to understand why various algorithms gave different answers for dimension. A further motivation came from the need to characterize more fully fractals with nonuniform measure. These sets are sometimes called multifractals and are characterized by an a priori measure that differs from the Hausdorff measure Γ[18]. Thus rather than measure just one dimension, one can compute the full spectrum of generalized dimensions from $D_{-\infty}$ to D_{∞}.

The formalism of coarse-grained measure introduced in Hausdorff's definition of dimension can be generalized to

$$\Gamma(A, D, r) = \left\{ \begin{array}{ll} \inf \Sigma_i \mu_i^q \delta_i^{(1-q)D} & q < 1 \\ \sup \Sigma_i \mu_i^q \delta_i^{(1-q)D} & q > 1 \end{array} \right\} \tag{23}$$

As before,

$$\lim_{r\to 0} \Gamma_q(A) = \begin{cases} 0 & (1-q)D < (1-q)D_q \\ \infty & (1-q)D > (1-q)D_q \end{cases} \qquad (24)$$

The transition between $\Gamma \to 0$ and $\Gamma \to \infty$ defines the generalized dimension:

$$D_q = \frac{1}{1-q} \inf \left\{ (1-q)D : \lim_{r\to 0} \Gamma_q(A, D, r) = 0 \right\} \qquad (25)$$

2.2.1.3 Information Dimension

As an alternative to the scaling of mass with size, one can also think of the dimension of a set in terms of how many real numbers are needed to specify a point on that set. In this case, dimension is something that counts the number of degrees of freedom. For sets more complicated than lines, surfaces and volumes, however, this informal definition of dimension needs to be extended.

One way to extend this definition is to determine not how many real numbers but how many bits of information are needed to specify a point to a given accuracy.

In general, $S(r) = -d\ log_2(r)$ bits of information are needed to specify the position of a unit d-dimensional hypercube to an accuracy of r. This leads to a natural definition for the information dimension of a set; it is given by the small r limit of $-S(r)/log_2(r)$, where $S(r)$ is the information (in bits) needed to specify a point on the set to an accuracy r. If $S(r)$ is the entropy, then $2^{S(r)}$ is the total number of available states, and $2^{-S(r)}$ can be interpreted as the average bulk of each state. This interpretation permits one to express the information dimension as a scaling of bulk with size.

Consider partitioning the fractal into boxes $\mathcal{B}_i$ of size r. To specify the position of a point to an accuracy r requires that one specify in which box the point is. The average information needed to specify one box is given by Shannon's formula:

$$S(r) = -\sum_i P_i \log_2 P_i \tag{26}$$

where P_i is the probability measure of the i^{th} box Pi = $\mu(\mathcal{B}_i)/\mu(\mathcal{A})$. This relation leads directly to an expression for the information dimension of the attractor:

$$D_I = \lim_{r\to 0} \frac{-S(r)}{\log_2(r)} = \lim_{r\to 0} \sum_i \frac{P_i \log_2 P_i}{\log_2(r)} \tag{27}$$

A generalized information measure[17,20] can be defined as

$$S_q = \frac{1}{q-1} \log \sum_i P_i^q \tag{28}$$

which reduces the Shannon's formula in the limit $q \to 1$. Thus

$$D_q = \lim_{r\to 0} \frac{-S(r)}{\log_2 r} = \frac{1}{q-1} \lim_{r\to 0} \log \sum_i \frac{P_i^q}{\log_2 r} \tag{29}$$

which is the generalized dimension.

2.3 Estimation of Fractal Dimension

It is emphasised that numerical techniques can only estimate the dimension of a fractal. Practical estimation of dimension begins with a definite description of the fractal object.

The box-counting dimension has a number of practical limitations[25], particularly at a high embedding dimension, and so a variety of other algorithms have also been developed.

The most popular way to compute dimension is to use the correlation algorithm, which estimates dimension based on the statistics of pointwise distances. The box-counting algorithm and the correlation algorithm are both in the class of fixed-size algorithms because they are based on the scaling of mass with size for fixed-size balls (or grids). An alternative approach uses fixed-mass balls, usually by looking at the statistics of distances to the k^{th} nearest neighbors. Both fixed-size and fixed-mass algorithms can be applied to estimation of generalized dimension D_q, although fixed size algorithms do not work well for $q < 1$ [24].

The intrinsic dimension is an integer dimension that provides an upper bound on the fractal dimension of the attractor by looking for the lowest dimensional manifold that can (at least locally) confine the data.

A generalized dimension is useful for quantifying the nonuniformity of the fractal, or, in general, for characterizing its multifractal properties. And this use is important if one wants to compare an exact and predictive theory with an experimental result. The goal of dimension estimation is often more qualitative in nature.

2.3.1 Average Pointwise Mass Algorithms

The pointwise mass function is defined as

$$B_x(r) = \mu[B_x(r)] \tag{30}$$

where, $B_x(r)$ = ball of radius r about the point x.

The scaling of $B_x(r)$ with r is what defines the pointwise dimension at x. If this scaling is the same for all x, then the fractal is uniform, and it follows that D_q is a constant independent of q and has the value of the pointwise dimension. For most fractals, however, pointwise dimension is not a global quantity, and some averaging is necessary for the quantity to be made global.

An estimate of the global pointwise dimension can be obtained by averaging the local pointwise dimension over a finite sample of points. This in principle gives the information dimension D_1. It is probably more efficient to find the limit of the average of the pointwise dimension as $r \rightarrow 0$ than to find the average of the limit of

$$D_q(x) = \lim_{r \to 0} \frac{\log B_x(r)}{\log r} \tag{31}$$

2.3.1.1 Correlation Dimension

A correlation integral is obtained by the arithmetic average of the pointwise mass function[21,22,78].

$$C(r) = \langle B_x(r) \rangle \tag{32}$$

which leads to the correlation dimension ν,

$$\nu = \lim_{r \to 0} \frac{\log C(r)}{\log r} \tag{33}$$

Here, $B_x(r)$ can be approximated from a finite data set of size N by

$$B_{x_j} \approx \frac{\{x_i : i \neq j \wedge \|x_i - y_j\| \leq r\}}{N-1} \tag{34}$$

$$B_{x_j} = \frac{1}{N-1} \sum_{i=1, i \neq j}^{N} \theta(r - \|x_i - y_j\|) \tag{.35}$$

where θ is the Heaviside step function:

$$\theta(x) = \begin{Bmatrix} 0 & x < 0 \\ 1 & x \geq 0 \end{Bmatrix} \tag{36}$$

Thus

$$C(n,r) = \frac{1}{N}\sum_{j=1}^{N} B_{x_j}(r) \tag{37}$$
$$= \frac{1}{N(N-1)}\sum_{i\neq j}\theta(r-\|x_i - y_j\|)$$
$$= \frac{\#\text{ of distances less than } r}{\#\text{ of distances altogether}}$$

Thus the correlation algorithm provides an estimate of dimension based solely on the statistics of pairwise distances.

For N points, $C(N,r)$ has a dynamic range of $O(N^2)$. Logarithmically speaking, this range is twice that available to $\tilde{n}(N,r)$. in the box-counting method. It is also twice the range available in an estimate of the pointwise mass function $B_x(r)$ for a single point x. This greater range is the one advantage that the correlation integral has over the average pointwise dimension.

2.3.1.2 Generalized Dimension from Averaged Pointwise Mass

A more general average than the direct arithmetic average used above is given by

$$G_q(r) = \left[\left\langle B_x(r)^{q-1}\right\rangle\right]^{\frac{1}{q-1}} \tag{38}$$

The scaling of this average with given the generalized dimension:

$$G_q(r) \sim r^{D_q} \tag{39}$$

$$D_q = \lim_{r\to 0} \frac{\log G_q(r)}{\log(r)} \tag{40}$$

where,

q=2 then a direct arithmetic average is obtained $\rightarrow \nu$

q=1 then an average is obtained $\rightarrow$ information dimension

and,

$$G_1(r) = \lim_{q\to 0} G_q(r) = e^{\langle \log B_x(r)\rangle} \tag{41}$$

is a geometric average of the pointwise mass function.

From a finite set of points, $G_q(N,r)$ can be approximated by

$$G_q(N,r) = \left\{\frac{1}{N}\sum_{i=1}^{N}\left[\frac{1}{N-1}\sum_{j=1,j\neq i}^{N}\theta(r-\|x_i-y_j\|)\right]^{q-1}\right\}^{\frac{1}{q-1}} \tag{42}$$

when q=2

$$G_2(N,r) = \frac{1}{N(N-1)} \sum_{i \neq j} \theta(r - \|x_i - y_j\|) = C(N,r) \tag{43}$$

2.3.1.3 q-Point Correlation

A q-point correlation integral defined by counting q-tuples of points that have the property that every pair of points in the q-tuple is separated by a distance of less than r:

$$C_q(N,r) = \frac{1}{N_q} \#\{(x_i, \ldots, x_{i_q}) : \|x_n - x_m\| \leq r, \forall n, m \in \{i, \ldots, i_q\}\} \tag{44}$$

which has the scaling behavior

$$C_q(r) \sim r^{(q-1)D_q} \tag{45}$$

$$D_q = \frac{1}{q-1} \lim_{r \to 0} \lim_{N \to \infty} \frac{\log C_q(N,r)}{\log r} \tag{46}$$

This equation can in principle be applied to all integers $q \geq 2$, although its implementation is awkward for q ≥ 3. since the number of q-tuples N_q grows very rapidly with N.

2.3.1.4 $\vec{k}$ Nearest Neighbor (Fixed Mass) Algorithms

The nearest-neighbor algorithms consider the scaling of sizes in fixed-mass balls.

The average distance to the k^{th} nearest neighbor $< r_k >$ is computed as a function of k. Let $R\left(\vec{x_i}, k\right)$ denote the distance between point $\vec{x_i}$ and its k^{th} nearest neighbor.

$$\langle r_k \rangle = \frac{1}{N} \sum_{i=1}^{N} R\left(\vec{x_i}, k\right) \tag{47}$$

The scaling $< r_k > \sim k^{1/D}$ defines the dimension. To estimate the Hausdorff dimension the moments of the average distance to the k^{th} nearest neighbor is used[5,23],

$$\langle r_k^{\nu} \rangle \sim \left[\frac{\Gamma(k+\nu/D)}{\Gamma(k) k^{\nu/D}} \right] \left(\frac{k}{N} \right)^{\nu/D} \tag{48}$$

2.3.2 Spectral Methods

Fourier spectral analysis is a widely accepted approach to the quantization of the fractal dimension.

2.3.2.1 One Dimensional

For a self-affine fractal the spectral energy density S of the profile must have a power law dependence on wave number k[44]:

$$S(k) \sim k^{-\beta} \tag{49}$$

Let $h(x)$ be a profile. The variance V is defined by

$$V(L) = \frac{1}{L}\int_0^L \left(h(x) - \bar{h}\right)^2 dx \tag{50}$$

where,

$L \equiv$ lenght over which profile is specified
$\bar{h} \equiv$ mean profile quantity

A necessary condition that the profile be a fractal is that the variance $V(L)$ must have a power-law dependence on the length L[83,84]

$$V(L) \sim L^{2H} \tag{51}$$

The standard deviation is related to the length of the profile L by

$$\delta(L) = [V(L)]^{\frac{1}{2}} \sim L^{H} \tag{52}$$

For fractional Brownian motion, $0 < H < 1$.

In order to define the relevant fractal dimension, we introduce a reference box with a width L and a height δ[83]. If fractal were self-similar, the box would be square; however, for self-affine fractals, arbitrary rectangular boxes must be used.

Consider a set of n^{th} order smaller boxes with width $L_n = L/n$ and height $h_n = \sigma/n$; n is an integer. The number of n^{th}-order boxes N_n required to cover a width L and a height $\sigma_n = \sigma(L/n)$ is

$$N_n = \frac{L\sigma_n}{L_n h_n} = n^2 \frac{\sigma_n}{\sigma} \tag{53}$$

using

$$\sigma(L) = [V(L)]^{\frac{1}{2}} \sim L^H \tag{54}$$

we obtain

$$\frac{\sigma_n}{\sigma} = \frac{\sigma(L/n)}{\sigma(L)} = \frac{1}{n^H} \tag{55}$$

from which

$$\mu_n = n^2 \frac{\sigma_n}{\sigma} = n^{2-H} = \left(\frac{L}{L_n}\right)^{2-H} \tag{56}$$

Since

$$N_i = \frac{C}{r_i^D} \tag{57}$$

then the fractal dimension is then

$$D = 2 - H \tag{58}$$

The spectral density $S(L)$ will also have a power-law dependence on the length L and can be related to the variance $V(L)$ by

$$S(L) = LV(L) \sim L^{\beta} \sim L^{1+2H} \tag{59}$$

so that

$$D = \frac{5-\beta}{2} \tag{60}$$

For a one-dimensional profile, D is expected to lie in the range $1 < D < 2$; the corresponding range for β is $1 < \beta < 3$.

2.3.2.2 Two Dimensional

The Fourier spectral approach of fractal analysis for one-dimensional profiles can be extended to two dimensional analysis.

An L x L image is subdivided into an N x N grid. The N^2 data points are denoted by h_{nm}, with n and m specifying the position in the x and the y directions, respectively.

The first step is to carry out a two dimensional discrete Fourier transform on the N^2 set of h_{nm} data points. An N x N array of complex coefficients H_{st} is obtained by the usual definition:

$$H_{st} = \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} h_{nm} \exp\left[-\frac{2\pi i}{N}(sn + tm)\right] \tag{61}$$

where s denotes the transform in the x direction (s = 0, 1, 2, ..., N-1) and t denotes the transform in the y direction (t = 0, 1, 2, ..., N-1). Then each transform coefficient H_{st} is assigned an equivalent radial number by using the relation

$$r = (s^2 + t^2)^{\frac{1}{2}} \tag{62}$$

The two dimensional mean spectral density S_2 for each radial wave number k_j is given by

$$S_{2j} = \frac{L^2}{N_j} \sum_{1}^{N_j} |H_{st}|^2 \tag{63}$$

where N_j is the number of coefficients that satisfy the condition $j < r < j+1$ and the summation is carried out over the coefficients H_{st} in this range.

The two dimensional equivalent of the one dimensional power law dependence on wave number is

$$S_{2j} \sim k_j^{-1-\beta} \tag{64}$$

The additional power of k_j is required because of the radial coordinates that are used in phase space.

The fractal dimension is given in the 2-D by

$$D_2 = 3 - H \tag{65}$$

For the spectral energy density we have

$$S_2(L) \sim L^2 V(L) \sim L^{(1+\beta)} \sim L^{2+2H} \tag{66}$$

such that

$$D_2 = \frac{7-\beta}{2} \tag{67}$$

2.3.3 Covering Blanket

The covering blanket method[56] is a method that estimates the fractal dimension for two dimensional data such as images. The concept of a covering blanket refers to a construct by which the gray pixel value of the input image are bounded between two secondary gray scale functions; a lower, $L(i, j, scale)$, and an upper, $U(i, j, scale)$, gray level surfaces.

At a scale of zero

$$U(i, j, 0) = L(i, j, 0) = g(i, j) \tag{68}$$

and $g(i, j)$ is the input image.

For other scales the upper surface $U(i, j, scale)$ and lower surface $L(i, j, scale)$ are defined recursively as

$$U(i,j,\varepsilon+1) = \max\left\{U(i,j,\varepsilon)+1, \max_{k,m\in\eta}[U(k,m,\varepsilon)]\right\} \tag{69}$$

$$L(i,j,\varepsilon+1) = \min\left\{L(i,j,\varepsilon)+1, \min_{k,m\in\eta}[L(k,m,\varepsilon)]\right\} \tag{70}$$

where

$$\eta = \{(k,n) | \text{distance}[(k,m),(i,j)] \leq 1\} \tag{71}$$

The covering blanket is defined to be the gray scale band of thickness 2ε created by the two secondary functions. For a one-dimensional signal, the covering blanket bounds an area. For a two-dimensional signal, the surface area is the volume occupied by the blanket divided by 2ε. The area of a gray level surface within a bounding rectangle R is computed from the upper and lower gray level surfaces, by substracting point by point the lower surface from the upper surface and summing over the observation window R.

$$A(\varepsilon) = \frac{\Sigma_{i,j \in R}\, U(i,j,\varepsilon) - L(i,j,\varepsilon)}{2\varepsilon} = \frac{V(\varepsilon)}{2\varepsilon} \tag{72}$$

The estimation of the fractal dimension involves taking the logarithm of the gray-level surface area function (i.e., a function of scale) and fitting a line to it. If the upper and lower gray level surfaces are not always raised and lowered, respectively, the area of a flat surface (i.e., the region of constant value) would be zero for all scales, and a correct estimate of the fractal dimension could not be obtained. By raising and lowering the surfaces, one can compute a correct

estimate of the fractal dimension for regions of constant value as well as for regions of various roughness.

2.4 Conclusions

As can be seen above, there are many techniques for estimating the fractal dimension of an image. Each year there is usually several new techniques being published. All share a common basis, i.e., they are essentially a density measure of a feature, mostly a point, sometimes an area, with respect to the change in the scale of the feature being used.

Because they share the same basis of measurement, they also share the same disadvantage. The fractal dimension is essentially a statistical measure of feature density with respect to scale. No single statistical measure can uniquely characterize a complex fractal texture. It would be like trying to differentiate between people faces by using the mean facial reflectance, or some such. A single statiscal measure cannot uniquely characterize a complex texture.

Several techniques have been proposed to compensate for this deficiency, to reduce classification ambiguity by providing more information about the statistics of texture complexity. Several of such techniques are the fractal signature[57],

multifractals[53], and lacunarity measurements[47]. A relatively novel technique for estimating texture is to use space filling scans[47,51] and lacunarity[47].

The fractal signature is the evolution of a fractal measure as the resolution or sampling density increases. The fractal signature shows anomalies at sharp boundaries and edges. Anomalies are characterized by a fractal dimension (at some scale) that is less than the topological dimension. Since man made objects have less fractal properties than natural textures, the fractal signature of man made objects will reflect smoothness at small scales and roughness at higher scales and thus changes rapidly with scale, whereas the signature of fractal signature of natural objects are relatively constant or change slowly with respect to scale. This implies that the fractal signature can be used for the detection of man made objects embedded in natural clutter.

Multifractals are the second type of modification made to the fractal measure to reduce classification ambiguities. The fractal measure is a global statistic of a texture. Multifractal is essentially a histogram of fractal measures of segmented regions of a texture. The texture is subdivided into a set of nonoverlapping segments which covers the texture region. Within each region a fractal measure is obtained. This provide more information if the local and global statistics of the texture differs, i.e. the texture is heterogeneous.

Lacunarity is not per se a measure of fractal dimension, which is more of a measure of roughness, but provide additional information about texture[47]. The lacunarity is a measure of the empty space within the texture. The use of lacunarity in conjunction with a fractal measure can reduce but not remove classification ambiguities.

Since those techniques do provide more information about the texture, they provide a more accurate way of identifying textures. But the added information provided by those technique do not seem sufficient to uniquely characterize textures. This is because whatever method is used to estimate the fractal dimension, they all measure the evolution of the incidence of a feature, in this many case either a point, or a line, with respect to the feature scale. There is not enough information content within a single fractal measure or its evolution with respect to sampling scale to uniquely characterize textures.

3 Texture Classification

3.1 Overview

As stated in the introduction we will assume that a texture is defined by both deterministic and statistical features. Thus a characterization technique which includes both deterministic and statistical information can have sufficient information content to uniquely classify textures. This presupposes that the classification features are chosen in such a manner that the information they provide are sufficiently discriminatory.

It is also assumed that a given texture can be comprised of the superposition of subtextures. One can think of a texture as the superposition of transparencies, each transparency being comprised of a single distinct subtexture. For a given texture there is an infinite number of possible combination of subtexture which can reproduce a given texture. In this case, the goal is not recreate the texture, but to discriminate between texture class, thus the desired subtextures are related to the differences between textures. The set of subtextures desired for classification do not have to provide a complete basis for all texture classes since discrimination and not reconstruction is desired. It is assumed that each subtexture is comprised of a substructure embedded within the subtexture field, and that the substructure is unique to that subtexture. It is further assumed that the given texture is a fractal,

thus the subtextures are also fractals, and therefore each embedded substructure is also a fractal.

The approach used is to extract the discriminating fractal features which allows a unique classification of the substructure embedded within the subtexture, then obtain a scale and position invariant statistical measure of the incidence of the substructure within each of the subtexture. The statistical measure used is the fractal measure. The set of fractal measure of each of the subtexture provide a unique characterization of a given texture since the substructure embedded within each subtexture are selected to provide discrimination of the texture classes. The subtextures exist in one or more, but not all texture classes. This vector set of fractal measure is called a "Polyfractal Measure", to differentiate for the multifractal measure and fractal signature techniques.

The extraction of the discriminating features used to classify the embedded substructure within the subtextures is implemented using a feedforward neural network with backpropagation, adaptive, learning momentum. A set of wavelet functions are derived from the set of extracted classification features. The classification features are then detected from a simple thresholding of the wavelet transform of each of the wavelet function with a given unknown texture class. The set of classification features are then compared with a classification features pertaining to a given substructure. Each subtexture field then produces a

relationship between substructure incidence and the scaling of the wavelet function in the wavelet transform used to detect the substructure. The polyfractal is then obtained from that incidence relationship. The technique used in the detection and classification of the substructure embedded within the subtexture is called "N-Wavelet coding", to differentiate it from classification techniques which require the mapping of the texture onto a wavelet transform domain, and identifying differences in the coefficients of the wavelet function components. This type of classification using wavelet transform technique works by adjusting the wavelet function coefficient to maximize the match between it and the unknown class. The finer differences between the unknown class and the wavelet function are reduced by adding to the wavelet function smaller scaled copies of itself, thus as smaller scaled version of its wavelet function is added the match between the sum of all scaled wavelet functions used and the unknown texture becomes arbitrarily small. The drawback in using this technique is that if the difference between two texture classes are small, those difference can be reflected by small differences across several wavelet coefficients instead of a large difference in one coefficient. If noise is present this will significantly decrease the classification performance. This problem occurs because an apriori selection of the wavelet function is made, and the wavelet function selected is not matched to the differences between texture classes. In the N-Wavelet coding technique, the wavelet function are derived from the differences between classes as reflected by the extracted discrimination features. It is shown that as the number, N, of wavelet functions used in the N-Wavelet coding technique

increases, for a given noise level, the probability of false alarm rate can decrease while the probability of detection remains constant. The noise compensation properties of the N-Wavelet coding technique stems from two reasons, the intrinsic properties of wavelet functions and the noise compensation produced by the N-Wavelet coding technique itself. Wavelet functions have two major constraints, they must have a zero moment, and they must decay at least geometrically with respect to extent. A typical zero moment constraint is a zero mean. If the extent of the wavelet function samples a large number of noise event, and if the wavelet function has a zero mean, the noise within the extent of the wavelet function will cancel out, assuming that the noise is random. The number, N, represents the number of classification features used, as N increases the classification redundancy increases, and the classification becomes more tolerant of noise.

Finally it will shown that by using a fractal scan, in this case a Peano scan, it is possible to map a 2D texture into 1D and still maintain sufficient adjacency information to uniquely characterize textures using 1D N-Wavelet coding, and polyfractals.

3.2 N-Wavelet Coding for Subtexture Identification

3.2.1 Introduction

A set of periodic continuous waveforms comprising six classes spanning a cross correlation coefficient between 0.68 to 0.99 is used to evaluate the N-Wavelet Coding technique. It is found that by increasing the length N of the N-Wavelet Coding it is possible to decrease the false alarm rate while maintaining a constant probability of detection.

A key drawback of many detection techniques, such as matched filters, correlation techniques, synthetic discriminant techniques, and a few others are that as the noise increases, to maintain the same probability of detection, the sensitivity is increased, producing an unwanted increase in the false alarm rate[15.80].

N-Wavelet Coding can produce a decrease in the false alarm rate for a constant probability of detection and noise level. The N-Wavelet Coding technique is also scale invariant and distortion tolerant.

3.2.2 Approach

An assumption in N-Wavelet coding is that the input data y_i of class i can be described as a linear superposition of M substructures g_j.

$$y_i = \sum_{i=1}^{M} c_{ij} g_j \tag{73}$$

where $c_{ij} = \{0,1\}$.

Thus the texture class i of the input data y_i can be described by a unique set $\{c_{ij}\}$ representing the presence or absence of substructure g_j in y_i.

$$c_{ij} = \{(y_i \bullet g_j) > \tau_j\} \tag{74}$$

where τ_j is the detection threshold for substructure j.

The substructure g_j comprising the universe of all possible textures are determined by a unique set $\{z_{jk}\}$ representing the absence or presence of classification features f_k.

A neural network is used to identity the features f_k used for detection and classification of substructures g_j. The artificial neural network is comprised of one or more layers, each layer is in turn comprised of neural nodes and their associated connection weights[71].

Each neural node k has nonlinear transfer function T_k. In this thesis a step function is assumed. A neural node k is activated if the inner product of its connection weights and the input data exceeds an activation threshold θ_k.

The magnitude of the inner product can be interpreted as a measure of similarity between its arguments. Another interpretation of the inner product is that it is a zero displacement correlation between two vector arguments. In this case the inner product of the neural node k weights with the input data y_i can be said to be a zero displacement correlation between the input data g_i and the reference function f_k represented by the weights of the neural node k.

Based on the above arguments it can be concluded that each neural node k detects a feature f_k which is equivalent to the weights of the neural node k. Those detected features f_k are then subsequently used by the next highest neural layers for classification purpose.

The number of nodes in the first (input) neural layer is equal the maximum number N, of features which can be extracted. The number of nodes in the first layer is equal to the number N, of extracted features only if the resulting weights of the input layer are not correlated.

$$f = \{f_1 \quad f_2 \quad \ldots \quad f_N\} \tag{75}$$

$$z_{jk} = [(g_j \bullet f_k) > \theta_k] \tag{76}$$

where z_{jk} is a binary valued variable.

The extracted features f_k are transformed into wavelet functions w_k through a biasing and windowing operation. A wavelet function must satisfy two constraints, it must have a zero moment and its magnitude must exhibit at least geometric decay with extent[10].

$$w_k = \left(\frac{\max(f_k \bullet g_j)}{\min\left(\left(f_k - \bar{f}_k\right) H_w \bullet g_j\right)} \right) \left(f_j - \bar{f}_j\right) H_w \tag{77}$$

The first expression in parenthesis is a scaling parameter used to ensure that the inner product of the wavelet function w_k with substructure g_j is at least as large as the inner product of the feature f_k and the substructure g_j. This allows setting the detection threshold θ_k equal to zero and simplifies the processing.

The second term in parenthesis is the biasing operation which ensures that one of the wavelet function moments, in this case the mean, is equal to zero.

The last term H_w is a windowing function used to constrain the wavelet function w_k to at least a geometric decay with extent.

$$H_w(r) = \cos^2\left(\frac{r-1}{R-1}\pi\right) \tag{78}$$

where $r = \{1,2, ..., R\}$, and R is the size of the window H_w and is equal to the size of the feature extracted by the neural network.

This completes the feature extraction process and wavelet function design. At this point a set of N wavelet functions has been obtained which can be used to detect a set of M substructures g_j to classify an input data as a class i texture.

Feature detection is implemented by using a wavelet transform u_k of the wavelet functions w_k and the input data y_i. The points where the wavelet transform exceeds a threshold set equal to zero, indicate that a feature f_k has been detected, those set of points produces a binary incidence vector v_k.

$$u_i = \{W[y_i, w_1] \quad W[y_i, w_2] \quad ... \quad W[y_i, w_N]\} \tag{79}$$

$$u_i = \{u_{i1} \quad u_{i2} \quad ... \quad u_{iN}\} \tag{80}$$

where the wavelet transform[72], W, and its inverse are defined by

$$u_{i1} = u_{i1}(a,b) \text{ for all } (a,b) \tag{81}$$

$$u_{i1}(a,b) = \int_{-\infty}^{\infty} y_i(x) w_1(x,a,b) dx \tag{82}$$

$$y_i(x) = \frac{1}{C} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} u_{i1}(a,b) w_1(x,a,b) da db \tag{83}$$

where $w_1(x,1,b)$, is the mother wavelet and

$$w_1(x,a,b) = \frac{1}{\sqrt{a}} w_1\left(\frac{x-b}{a}\right) \tag{84}$$

is the set of daughter wavelet, and

$$C = \int_{-\infty}^{\infty} \frac{|G(\omega)|^2}{|\omega|} d\omega < \infty, \qquad \text{where} \quad G(\omega) = \text{FT}[w_1(x)] \tag{85}$$

the wavelet transforms are thresholded with respect to zero,

$$\nu_i = \{u_i > 0\} = \{\nu_{i1} \quad \nu_{i2} \quad \ldots \quad \nu_{iN}\} \tag{86}$$

Thus, ν_i is binary vector which provide a unique identification of substructure g_j. The incidence measure vector obtained from the incidence of substructure g_j then uniquely characterize texture y_i.

The binary sequence ν_i is compared with the test sequence t_k and the minimum mismatch which is less than a threshold θ_k determines the classification of the substructure g_j.

$$p_i = \left(\min_k (N - (\nu_i \bullet t_k)) \le \theta_k\right) k \tag{87}$$

The value k of p_i is the classification identification number. If $k = 0$ then no detection has occurred, otherwise if $k \neq 0$ then detection and classification has not occurred.

The value of p_i, k is the classification id number and N is the number of wavelet functions used to execute the detection. If $k = 0$ then no detection has occurred, otherwise y_k has been detected.

If there is noise or distortion, the degree of mismatch must be less than the minimum Hamming distance minus one, times one-half between class k and all other classes l.

The relationship between N, the number of features used for classification and the probability missclassification P_m is derived next.

Let the probability of error P_ε, for a constant noise level, be given by the conditional probabilities,

$$\begin{aligned} P_\varepsilon &= q(\nu_{ik} = 1 | \nu_{ik} = 0) + q(\nu_{ik} = 0 | \nu_{ik} = 1) \\ &= 2q(\nu_{ik} = 1 | \nu_{ik} = 0) = 2q(\nu_{ik} = 0 | \nu_{ik} = 1) \end{aligned} \tag{88}$$

then the number of erroneous features detected or misclassified f_ε , if N wavelet functions are used, is given by,

$$f_\varepsilon = NP_\varepsilon \tag{89}$$

The number of features detected incorrectly must be sufficiently low to ensure detection, i.e. it must be less than an error threshold value τ_k before detection occurs. This error threshold value is directly proportional to the minimum Hamming distance, d_H, between a given class k and all other classes l.

$$\tau_k = \frac{d_H - 1}{2} = \frac{\min\limits_{l \neq k} [\Sigma (t_k \neq t_l)] - 1}{2} \tag{90}$$

Then the detection is given by substituting eq. (85) into eq. (82),

$$p_i = \left(\min_k \left(\sum^N (v_i \neq t_k) \right) \leq \frac{1}{2} \left\{ \min_{l \neq k} \left(\sum^N (t_k \neq t_l) \right) - 1 \right\} \right) k \tag{91}$$

$$p_i = \left(\min_k \left[N - \sum^N (v_i = = t_k) \right] \leq \frac{1}{2} \left\{ \min_{l \neq k} \left[N - \sum^N (t_k = = t_l) \right] - 1 \right\} \right) k \tag{92}$$

If the wavelet functions identifies orthogonal features, then

$$\sum^N (v_i = = t_k) = N\delta_{ik} - f_\varepsilon \qquad \text{where } f_\varepsilon < N \tag{93}$$

and

$$\sum_{k}^{N}(t_k == t_l) = N\delta_{kl} = 0 \qquad \text{since} \quad l \neq k \tag{94}$$

and we obtain,

$$p_i = \left(\min_k [N - (N\delta_{ik} - f_\varepsilon)] < \tfrac{1}{2}\left\{ \min_{l \neq k} [N - N\delta_{kl}] - 1 \right\}\right)k \tag{95}$$

$$p_i = \left(f_\varepsilon < \frac{N-1}{2}\right)i = \left(NP_\varepsilon < \frac{N-1}{2}\right)i = \left(P_\varepsilon < \frac{1}{2}\left(1 - \frac{1}{N}\right)\right)i \tag{96}$$

Thus, with orthogonal feature selection, detection occurs as long as P_ε is less than $(1 - 1/N)/2$. If the features used to do the detection are not orthogonal then from eq. 87,

$$p_i = \left(\min_k (N) - \max_k \left(\sum_{k}^{N}(v_i == t_k)\right) < \frac{1}{2}\left\{ \min_{l \neq k} (N) - \max_{l \neq k} \left(\sum_{k}^{N}(t_k == t_l)\right) - 1 \right\}\right) \tag{97}$$

$$p_i = \left(N - \max_k (N\delta_{ik} - f_\varepsilon) < \frac{1}{2}\left\{N - \max_{l \neq k} (S_{kl}) - 1\right\}\right)k \tag{98}$$

$$p_i = \left(f_\varepsilon < \frac{N - S_{kl} - 1}{2}\right) \tag{99}$$

where, $0 < S_{kl} < N$

then the detection is given by,

$$p_i = \left(P_\varepsilon < \frac{1}{2}\left\{1 - \frac{S_{kl}+1}{N}\right\}\right)k \tag{100}$$

Thus for a constant noise level, or a constant probability of error, if the number of features N used for characterizing the target is sufficiently high, and the intraclass differences are smaller than the interclass differences, detection will occur.

The probability of misclassification, P_m, is the probability that the number of missclassified features, f_ε , exceeds a threshold value related to the minimum Hamming, d_H , between a given class i and any other class j . In other words, misclassification can occur if

$$f_\varepsilon \rangle \frac{d_H - 1}{2} \tag{101}$$

and the probability of misclassification can be given by

$$P_m = \sum_{i=(d_H-1)/2}^{N} \binom{N}{i} p_\varepsilon^i (1-p_\varepsilon)^{N-i} \tag{102}$$

where

$$\binom{N}{i} = \frac{N!}{(N-i)!i!} \tag{103}$$

As can be seen in Table 1, as the Hamming distance, d_H, increases, for a fixed N, the probability of missclassification decreases. Also, as N increases for a fixed d_H, the probability of missclassification first increases then decreases. It will be shown that the Hamming distance, d_H, also increases with increasing N assuming that the correlation coefficient between classes is sufficiently small.

Let the correlation coefficient, r, be defined by

$$r = E\left[\frac{(\nu_i - \bar{\nu}_i)(\nu_j - \bar{\nu}_j)}{\sigma_i \sigma_j}\right] \tag{104}$$

where

$$\begin{aligned} \sigma_i^2 &= E\left[(\nu_i - \bar{\nu}_i)^2\right] \\ \sigma_j^2 &= E\left[(\nu_j - \bar{\nu}_j)^2\right] \end{aligned} \tag{105}$$

then

$$r = \frac{E[\nu_i \nu_j] - \bar{\nu}_i \bar{\nu}_j}{\sqrt{E[\nu_i^2] - \bar{\nu}_i^2}\,\sqrt{E\left[\nu_j^2\right] - \bar{\nu}_j^2}} \tag{106}$$

Let,

$$\nu_i, \nu_j = \{-1, 1\} \tag{107}$$

then

$$r = \frac{E[\nu_i \nu_j] - \bar{\nu}_i \bar{\nu}_j}{\sqrt{1-\bar{\nu}_i^2}\sqrt{1-\bar{\nu}_j^2}} \tag{108}$$

and

$$E[\nu_i \nu_j] = \frac{1}{N}\sum_{k=1}^{N} \nu_{ik}\nu_{jk} = \frac{1}{N}\langle \nu_i | \nu_j \rangle \tag{109}$$

$$\langle \nu_i | \nu_j \rangle = N - 2d_H \tag{110}$$

thus

$$r = \frac{1}{N}\frac{(N-2d_H) - N\bar{\nu}_i\bar{\nu}_j}{\sqrt{1-\bar{\nu}_i^2}\sqrt{1-\bar{\nu}_j^2}} \tag{111}$$

$$d_H = \frac{N}{2}\left\{1 - r\sqrt{1-\bar{\nu}_i^2}\sqrt{1-\bar{\nu}_j^2} - \bar{\nu}_i\bar{\nu}_j\right\} \tag{112}$$

If we assume that the probability of occurrence of features are equal then

$$p(v_i = -1) = p(v_i = 1) \qquad (113)$$
$$p(v_j = -1) = p(v_j = 1)$$

from which

$$\bar{v}_i, \bar{v}_j = 0 \qquad (114)$$

and

$$d_H = \frac{N}{2}(1 - r) \qquad (115)$$

from which the probability of missclassification can be stated as,

$$P_m(N, r, p_\varepsilon) = \sum_{i=\max[1,(1/2)(N(1-r)/2-1)]}^{N} \binom{N}{i} p_\varepsilon^i (1 - p_\varepsilon)^{N-i} \qquad (116)$$

The probability of missclassification can be reduced by increasing N if the following condition is met,

$$P_m(N+1, r, p_\varepsilon) \langle P_m(N, r, p_\varepsilon) \qquad (117)$$

It can be seen from eq. 117 and Table 2, that as the number of features used in the classification process is increased, the Hamming distance between classes increases also. The magnitude of this increase is inversely proportional to the correlation coefficient between classes.

3.2.3 Results

A six class one dimensional data input (Fig. 1) is used to evaluate the N-Wavelet Coding technique. The cross correlation coefficients of the six classes range between 0.68 to 0.99 (Table 3). The data input is used to train an artificial neural network. The neural network is a standard three layer with a step function as the neural node transfer function. The learning algorithm is a standard backpropagation with adaptive learning and momentum coefficients. The number of nodes are adjusted to obtain, 4,6,10,18 (Fig. 2-4) and 36 classification features. Subsequently, the primitive features are extracted through fundamental spectral analysis (Fig. 5-7). The primitive features are then used to produce the wavelelet functions (Fig. 8-10). The wavelet functions are used to obtain the classification vectors (Fig. 11).

A single scale wavelet transform of all the data input with all the wavelet functions is performed and then thresholded. The resulting feature vector is then

matched with test patterns to identify and classify each position in the input data (Fig. 12-18).

It is found that the detection always occurs but there are significant misclassification errors. It is also found that by increasing the number of wavelet functions used, the number of misclassification errors approaches zero.

The Hamming distance and the crosscorrelation coefficients between the classes are linked (eq. 113). For high crosscorrelation coefficients, the Hamming distance is small. As the number of wavelet functions used in the N-Wavelet Coding detection technique increases, the Hamming distance between the classes increases with the least correlated classes increasing the most (Table 3-6).

It is found that while detection occurs, there are significant false alarm errors. It is also found that by increasing the number of wavelet functions used, the false alarm rate can be made to approach zero (Fig. 19).

3.2.4 Conclusions

The N-Wavelet Coding detection/classification technique shows significant advantages, in that by increasing the number of wavelet functions used, it is possible to decrease the false alarm rate, while maintaining detection sensitivity

(Fig. 12-18, Table 2). The least correlated classes shows the most decrease in the false alarm rate from increasing the length of the N-Wavelet Coding (Table 3-6).

One can see the highly correlated classes as a distorted or noisy representations of each other, a small distortion producing a highly correlated signal, a large distortion producing a highly uncorrelated signal (Fig. 1). Since the N-Wavelet Coding can discriminate between 0.99 and 0.68 correlated classes, the technique can be said to be noise and distortion tolerant.

While the wavelet transform was performed at one scale, for the sake of simplicity, the N-wavelet coding technique can be generalized easily to multiple scales to produce scale invariant detection and classification. To demonstrate the concept a one dimensional case was used, but this technique is applicable to a two or more dimensional case.

The experimental results validate the theoretical predictions. In conclusion the N-Wavelet Coding shows the potential for being, a noise and distortion tolerant, and a scale invariant, detection and classification technique.

The characteristics of N-Wavelet Coding stems from the use of optimal custom designed wavelet functions for classification producing an inherent error

correction code. As the length of this wavelet code is increased, redundancy is increased, and the error correction can compensate for more errors or noise. The error correction code also decrease the false alarm rate. The limitation of this technique are the limits of the ability of the error correcting code to compensate for the errors produced by noise. The length of the N-Wavelet Code needed to meet detection probability and false alarm rate requirements may be so high that the classification speed may be unacceptably low. This is because the computational requirements increases with the code length.

N \ i	4	8	12	16	20	24	28	32	36	40
4	0.29									
8	0.38	0.15								
12	0.38	0.23	0.08							
16	0.33	0.27	0.14	0.05						
20	0.27	0.28	0.19	0.09	0.03					
24	0.21	0.27	0.22	0.13	0.06	0.02				
28	0.16	0.24	0.24	0.16	0.09	0.04	0.01			
32	0.12	0.21	0.23	0.19	0.12	0.06	0.02	0.01		
36	0.09	0.18	0.22	0.2	0.14	0.08	0.04	0.02	0	
40	0.06	0.14	0.2	0.21	0.16	0.11	0.06	0.03	0.01	0

Table 1. Probability of Misclassification ($p_e = 0.1$)

N \ i	4	8	12	16	20	24	28	32	36	40
4	0.29									
8	0.38	0.15								
12	0.38	0.23	0.08							
16	0.33	0.27	0.14	0.05						
20		0.28	0.19	0.09	0.03					
24			0.22	0.13	0.06	0.02				
28				0.16	0.09	0.04	0.01			
32				0.19	0.12	0.06	0.02	0.01		
36				0.2	0.14	0.08	0.04	0.02	0	
60					0.16	0.11	0.06	0.03	0.01	0

Table 2. Probability of Misclassification ($p_e = 0.1$, $r = 0.5$)

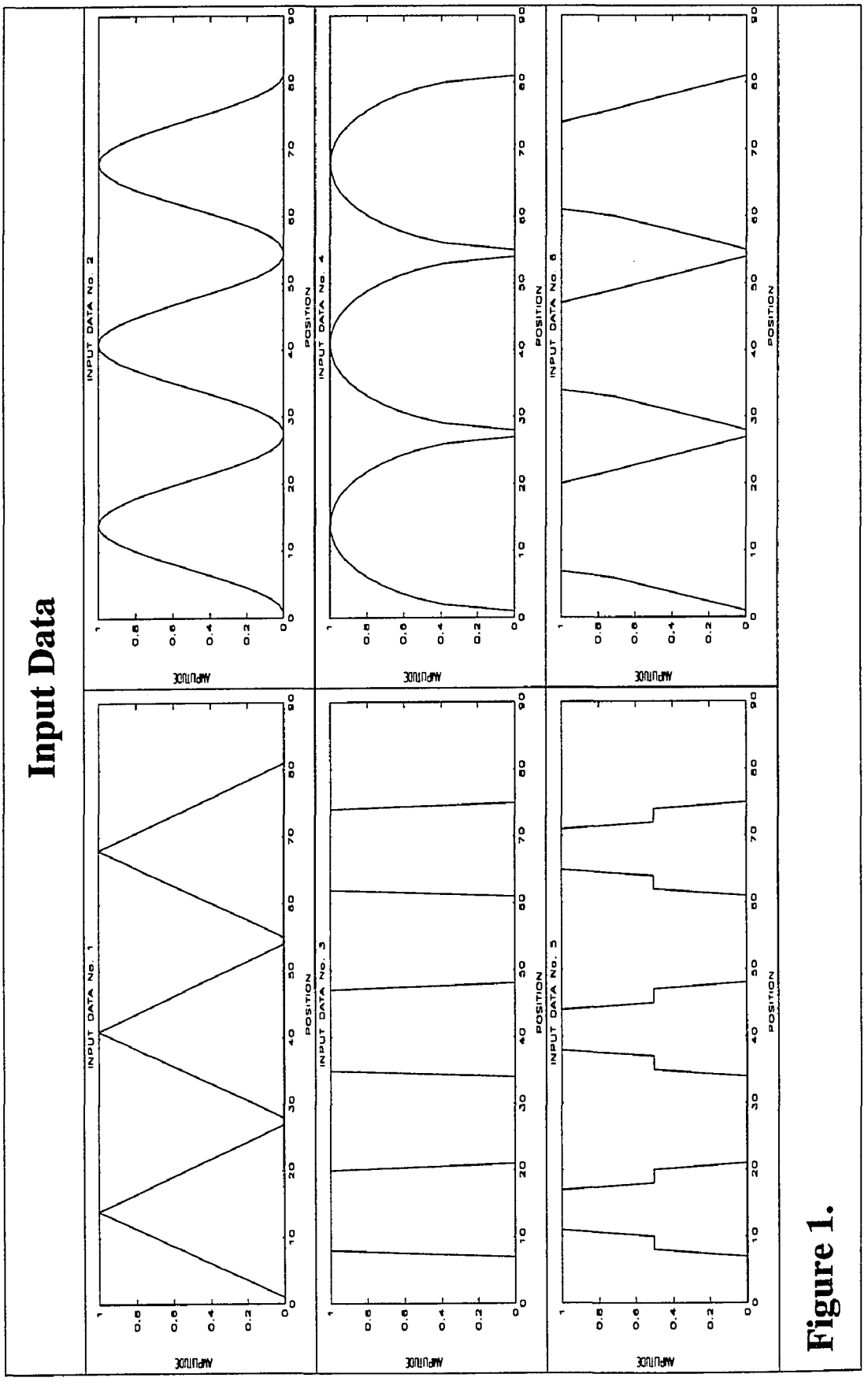
Input Data
INPUT DATA No. 1
INPUT DATA No. 2
INPUT DATA No. 3
INPUT DATA No. 4
INPUT DATA No. 5
INPUT DATA No. 6
AMPLITUDE
POSITION

Figure 1.

Cross Correlation Coefficient for Data Class 1-6

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	1.00000	0.99210	0.86671	0.88795	0.90989	0.90933
Class 2	0.99210	1.00000	0.90258	0.85980	0.92622	0.90701
Class 3	0.86671	0.90258	1.00000	0.71242	0.91192	0.78937
Class 4	0.88795	0.85980	0.71242	1.00000	0.88395	0.95311
Class 5	0.90989	0.92622	0.91192	0.88395	1.00000	0.71984
Class 6	0.90933	0.90701	0.78937	0.95311	0.71984	1.00000

Sonlinh Phuvan, 17 December 1992

Table 3.

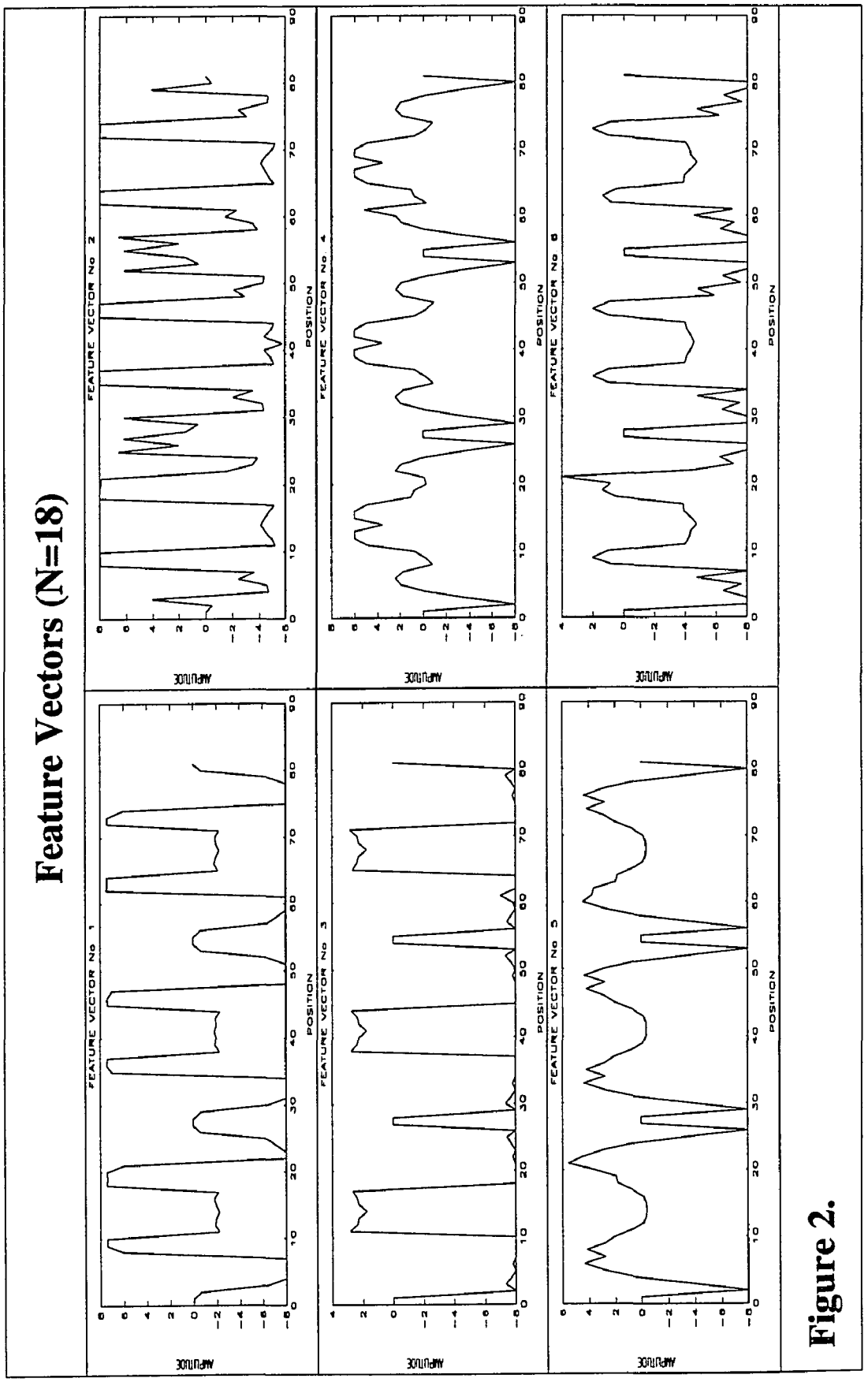
Feature Vectors (N=18)
FEATURE VECTOR No 1
FEATURE VECTOR No 2
FEATURE VECTOR No 3
FEATURE VECTOR No 4
FEATURE VECTOR No 5
FEATURE VECTOR No 6
AMPLITUDE
POSITION

Figure 2.

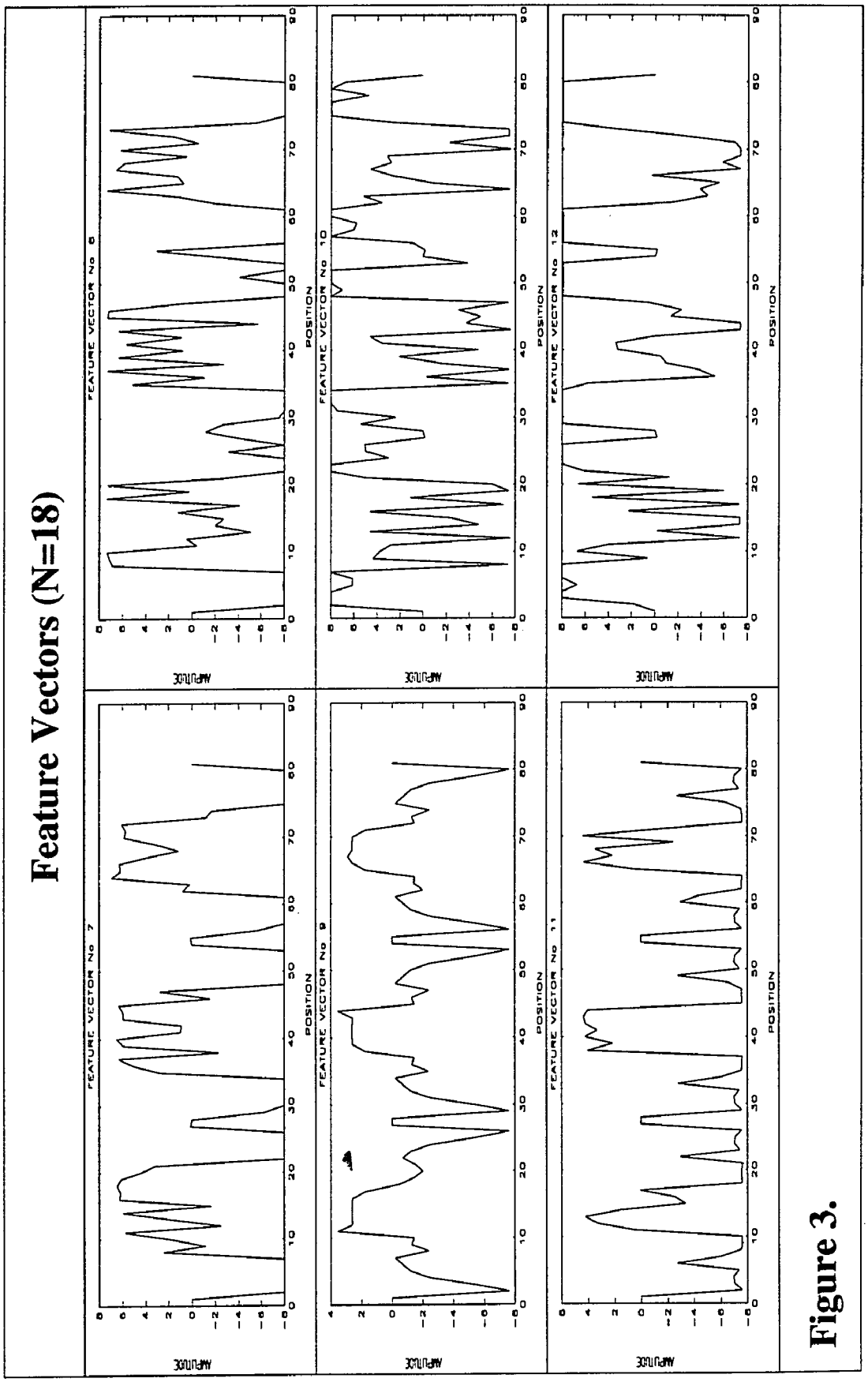
Feature Vectors (N=18)
FEATURE VECTOR No 6
FEATURE VECTOR No 7
FEATURE VECTOR No 8
FEATURE VECTOR No 9
FEATURE VECTOR No 10
FEATURE VECTOR No 11
FEATURE VECTOR No 12
AMPLITUDE
POSITION

Figure 3.

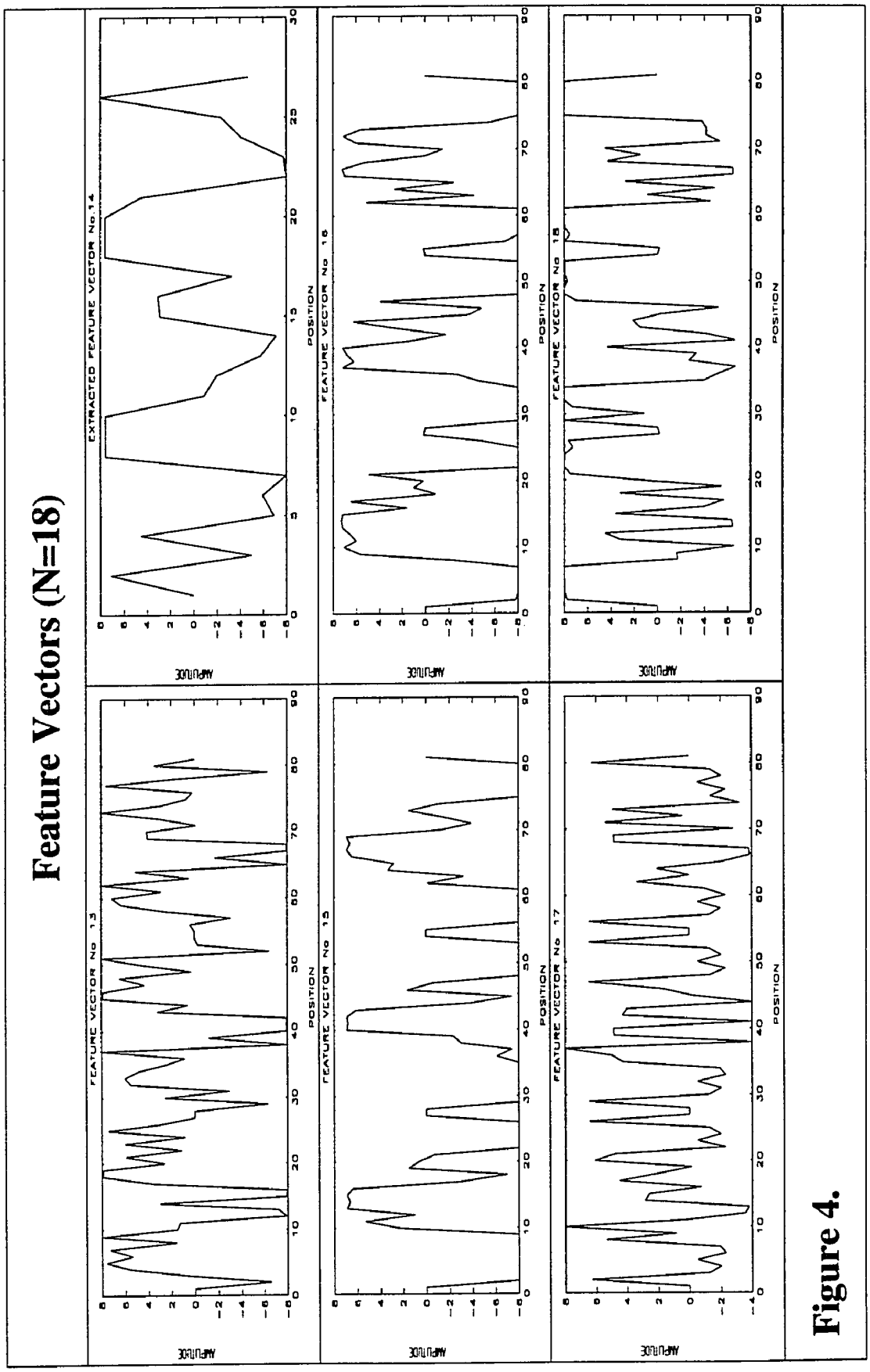
Feature Vectors (N=18)
EXTRACTED FEATURE VECTOR No.14
FEATURE VECTOR No 13
FEATURE VECTOR No 15
FEATURE VECTOR No 16
FEATURE VECTOR No 17
FEATURE VECTOR No 18
AMPLITUDE
POSITION

Figure 4.

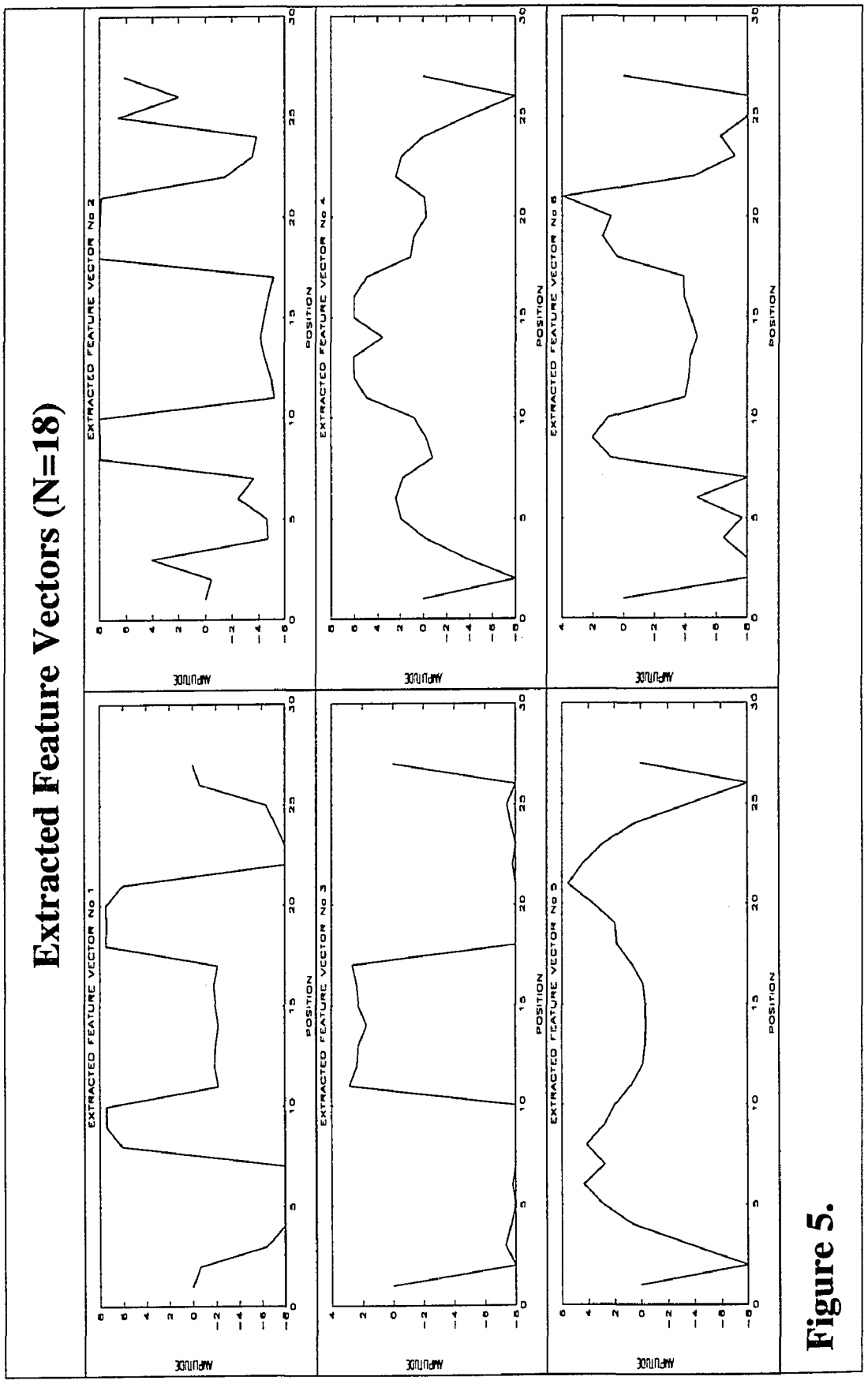
Extracted Feature Vectors (N=18)
EXTRACTED FEATURE VECTOR No 1
EXTRACTED FEATURE VECTOR No 2
EXTRACTED FEATURE VECTOR No 3
EXTRACTED FEATURE VECTOR No 4
EXTRACTED FEATURE VECTOR No 5
EXTRACTED FEATURE VECTOR No 6
AMPLITUDE
POSITION

Figure 5.

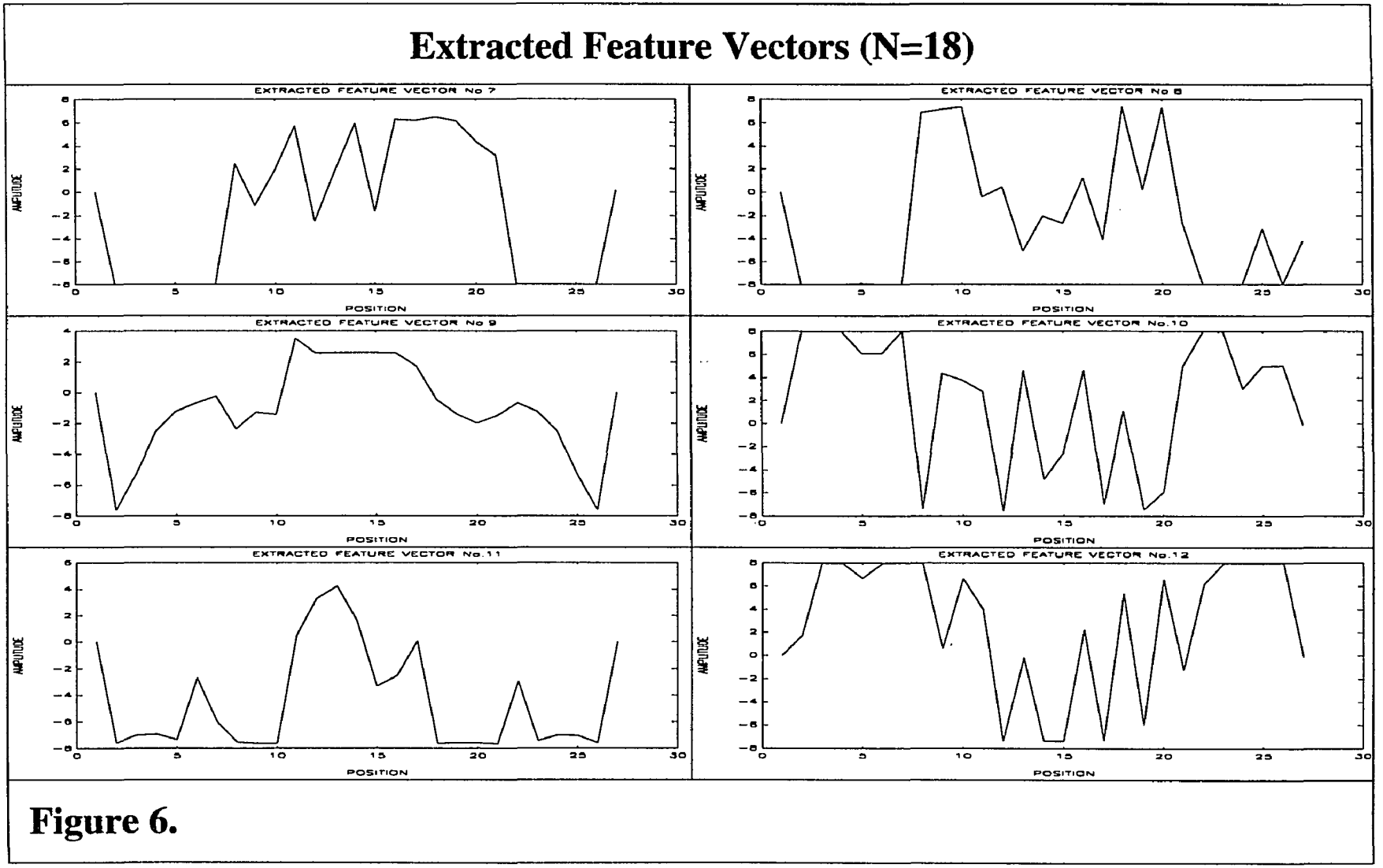
Extracted Feature Vectors (N=18)
EXTRACTED FEATURE VECTOR No 7
EXTRACTED FEATURE VECTOR No 8
EXTRACTED FEATURE VECTOR No 9
EXTRACTED FEATURE VECTOR No.10
EXTRACTED FEATURE VECTOR No.11
EXTRACTED FEATURE VECTOR No.12
AMPLITUDE
POSITION

Figure 6.

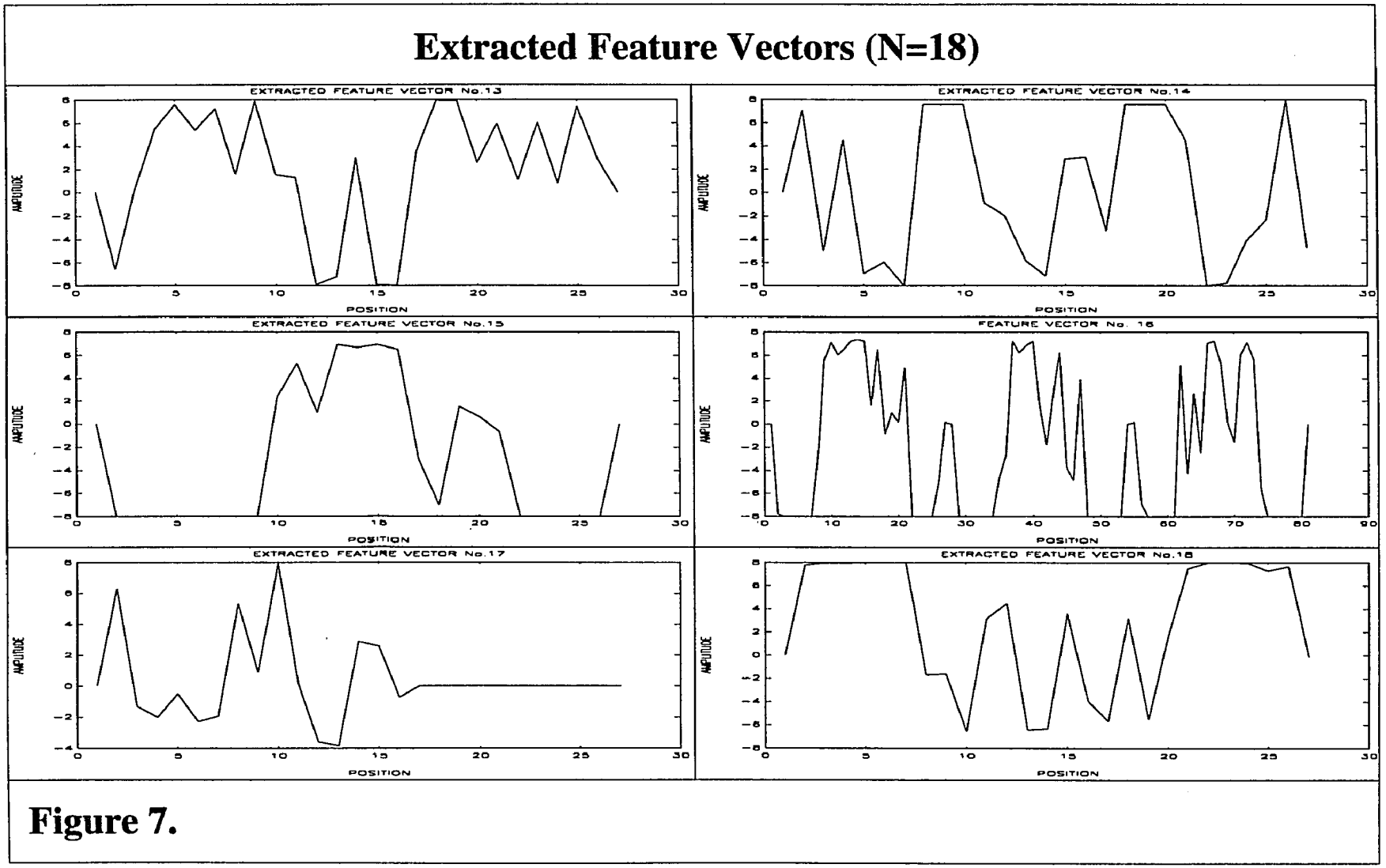
Extracted Feature Vectors (N=18)
EXTRACTED FEATURE VECTOR No.13
EXTRACTED FEATURE VECTOR No.14
EXTRACTED FEATURE VECTOR No.15
FEATURE VECTOR No. 16
EXTRACTED FEATURE VECTOR No.17
EXTRACTED FEATURE VECTOR No.18
AMPLITUDE
POSITION

Figure 7.

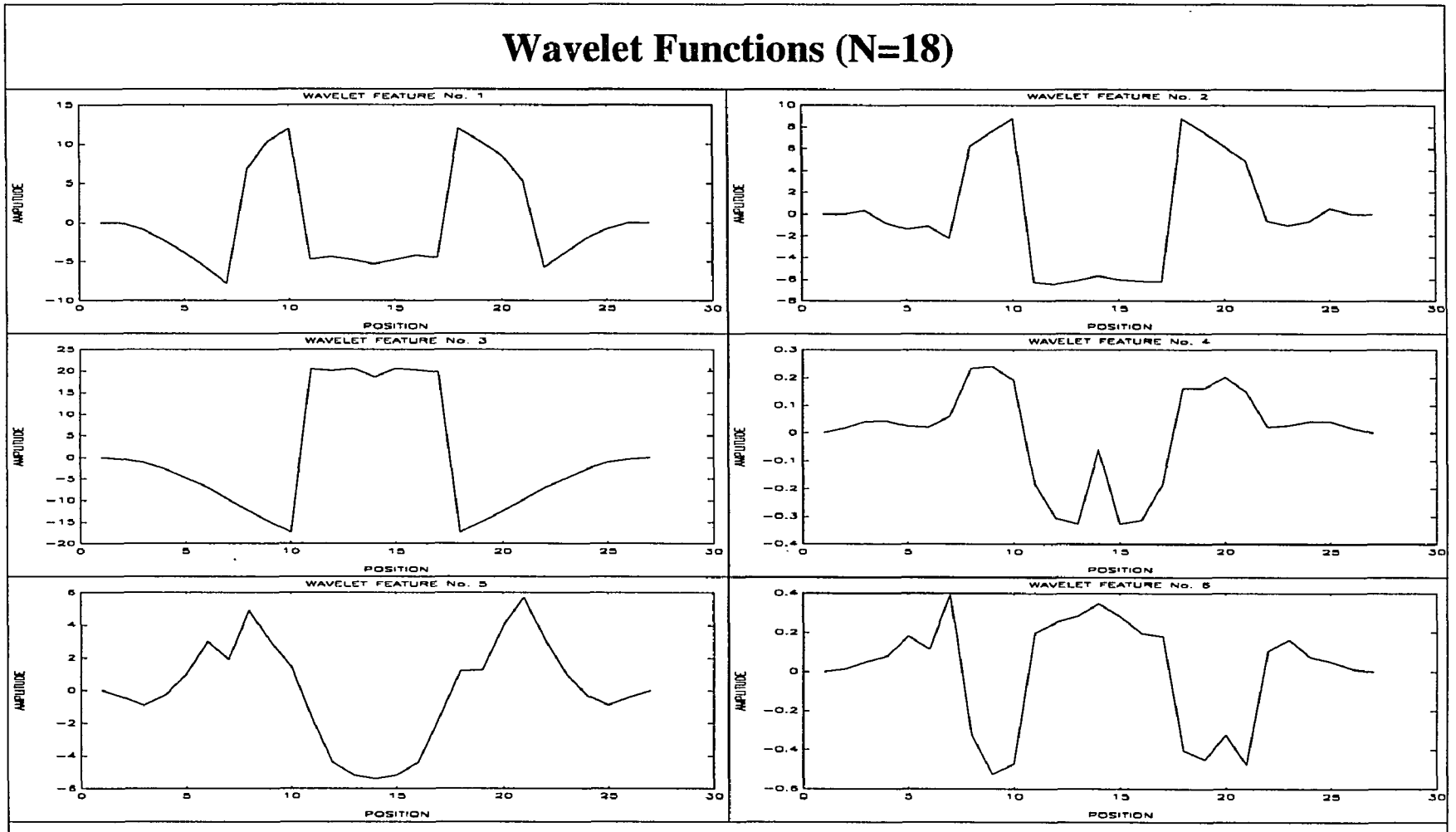
Wavelet Functions (N=18)
WAVELET FEATURE No. 1
WAVELET FEATURE No. 2
WAVELET FEATURE No. 3
WAVELET FEATURE No. 4
WAVELET FEATURE No. 5
WAVELET FEATURE No. 6
AMPLITUDE
POSITION

Figure 8.

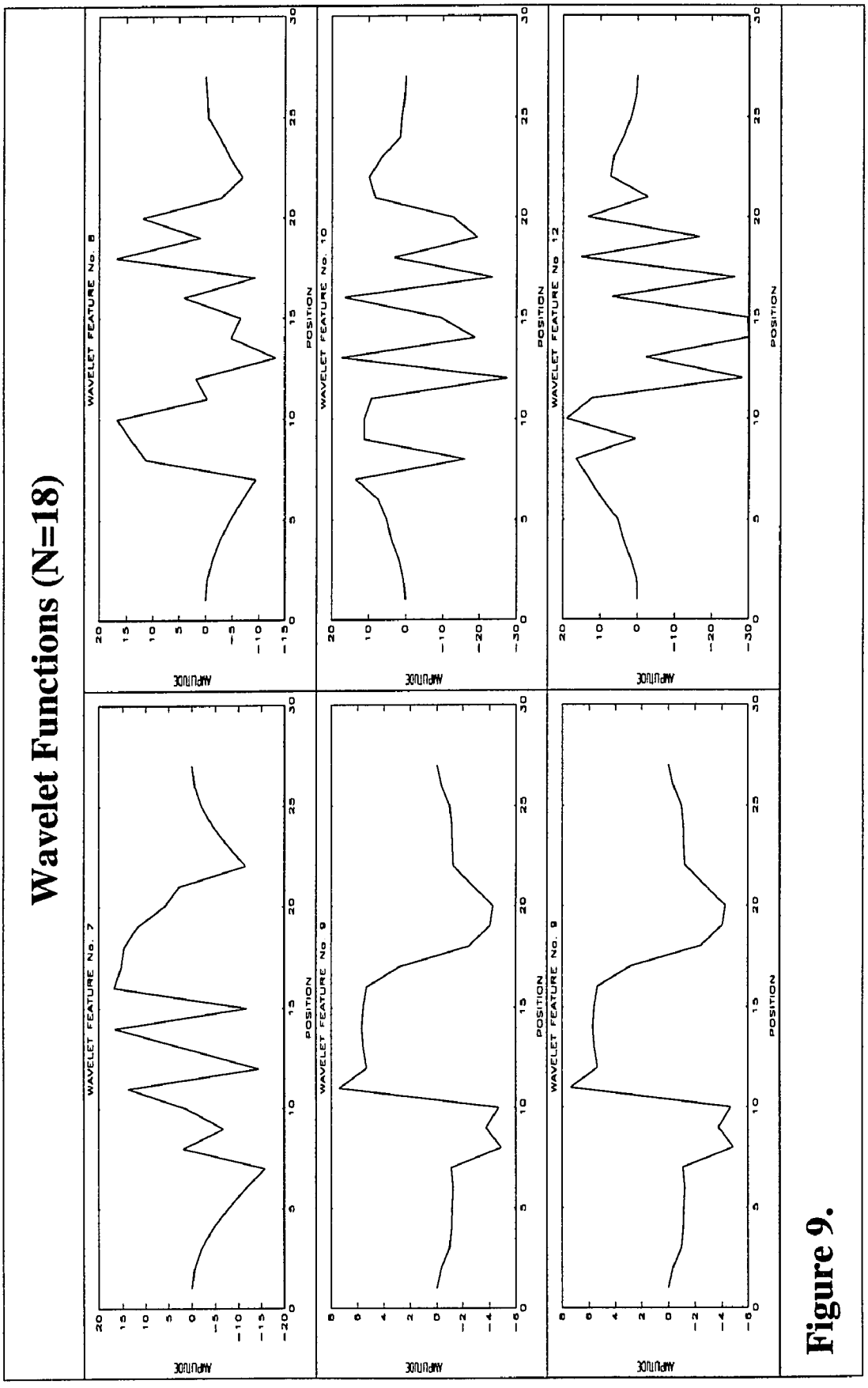
Wavelet Functions (N=18)
WAVELET FEATURE No. 7
WAVELET FEATURE No. 8
WAVELET FEATURE No. 9
WAVELET FEATURE No. 10
WAVELET FEATURE No. 11
WAVELET FEATURE No. 12
AMPLITUDE
POSITION

Figure 9.

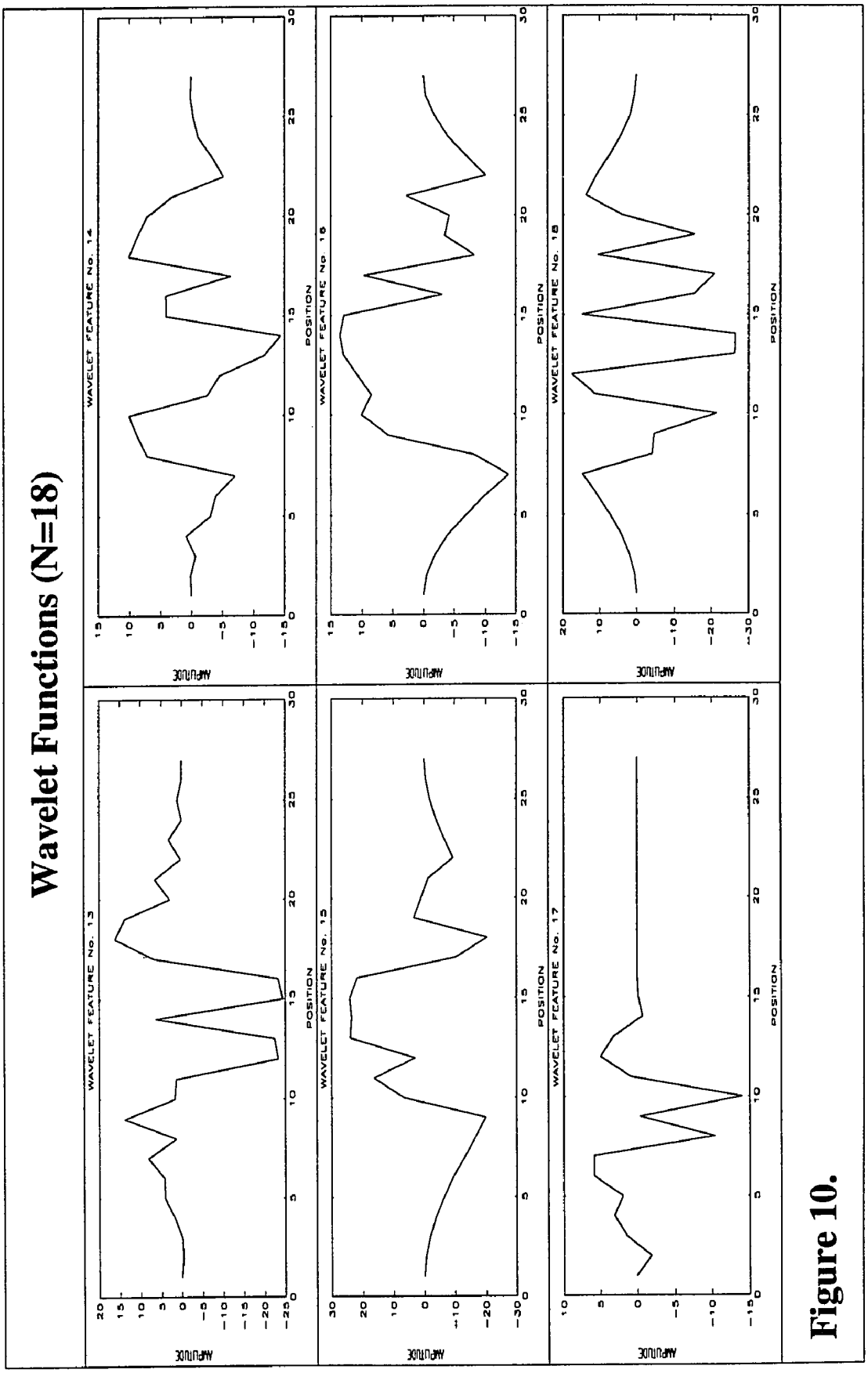
Wavelet Functions (N=18)
WAVELET FEATURE No. 13
WAVELET FEATURE No. 14
WAVELET FEATURE No. 15
WAVELET FEATURE No. 16
WAVELET FEATURE No. 17
WAVELET FEATURE No. 18
AMPLITUDE
POSITION

Figure 10.

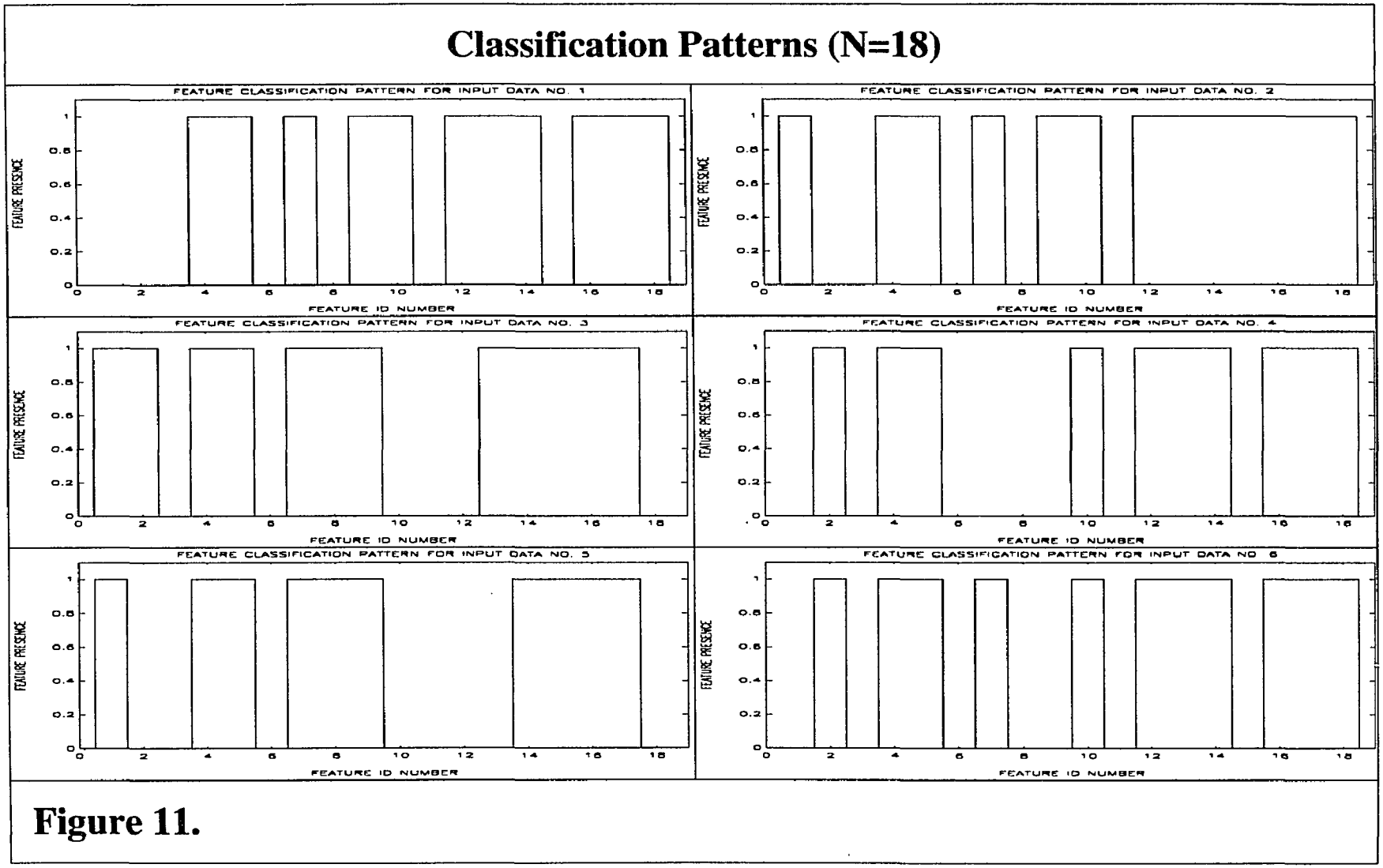
Classification Patterns (N=18)
FEATURE CLASSIFICATION PATTERN FOR INPUT DATA NO. 1
FEATURE CLASSIFICATION PATTERN FOR INPUT DATA NO. 2
FEATURE CLASSIFICATION PATTERN FOR INPUT DATA NO. 3
FEATURE CLASSIFICATION PATTERN FOR INPUT DATA NO. 4
FEATURE CLASSIFICATION PATTERN FOR INPUT DATA NO. 5
FEATURE CLASSIFICATION PATTERN FOR INPUT DATA NO. 6
FEATURE PRESENCE
FEATURE ID NUMBER

Figure 11.

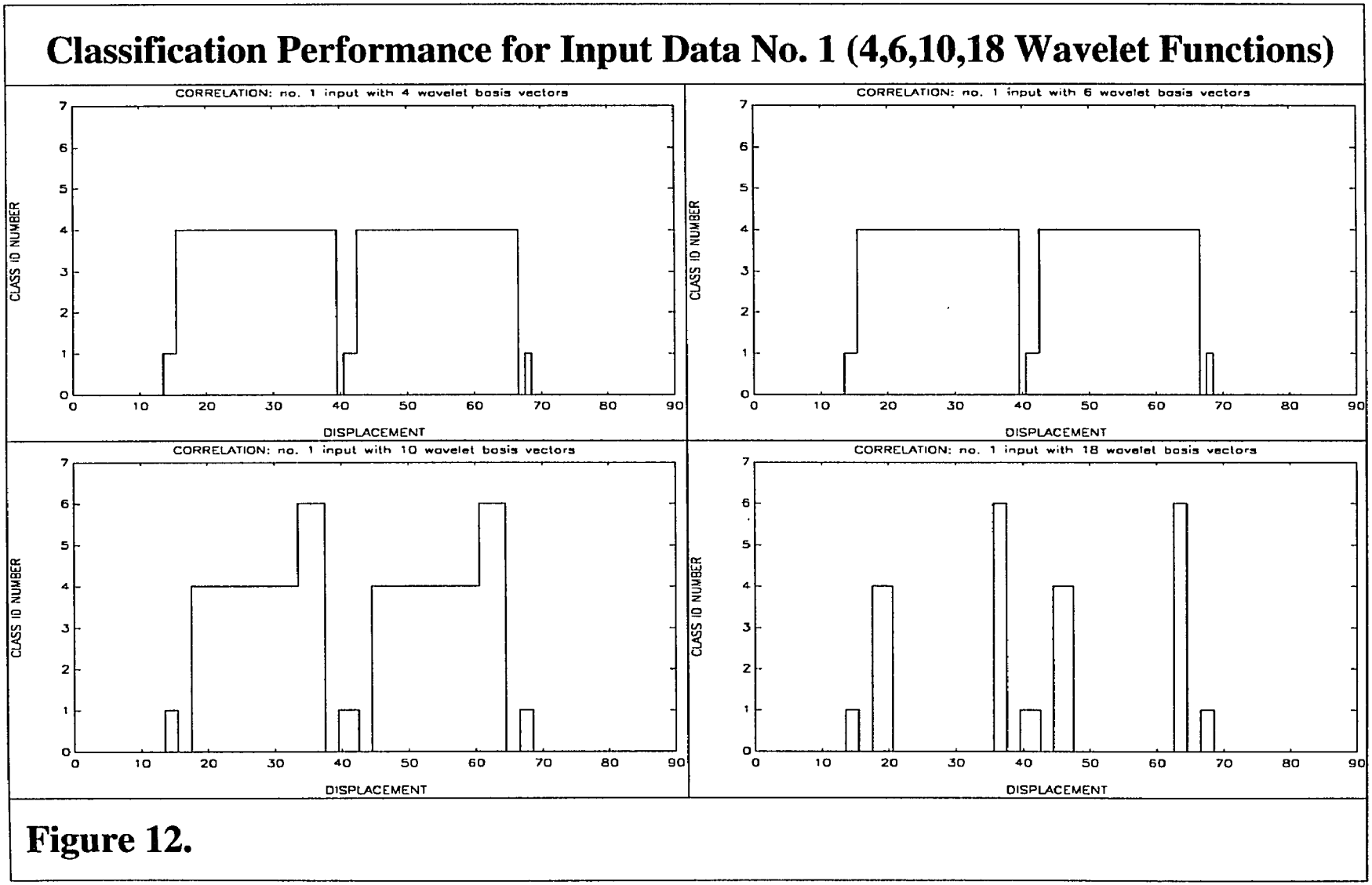
Classification Performance for Input Data No. 1 (4,6,10,18 Wavelet Functions)
CORRELATION: no. 1 input with 4 wavelet basis vectors
CORRELATION: no. 1 input with 6 wavelet basis vectors
CORRELATION: no. 1 input with 10 wavelet basis vectors
CORRELATION: no. 1 input with 18 wavelet basis vectors
CLASS ID NUMBER
DISPLACEMENT
0 1 2 3 4 5 6 7
0 10 20 30 40 50 60 70 80 90

Figure 12.

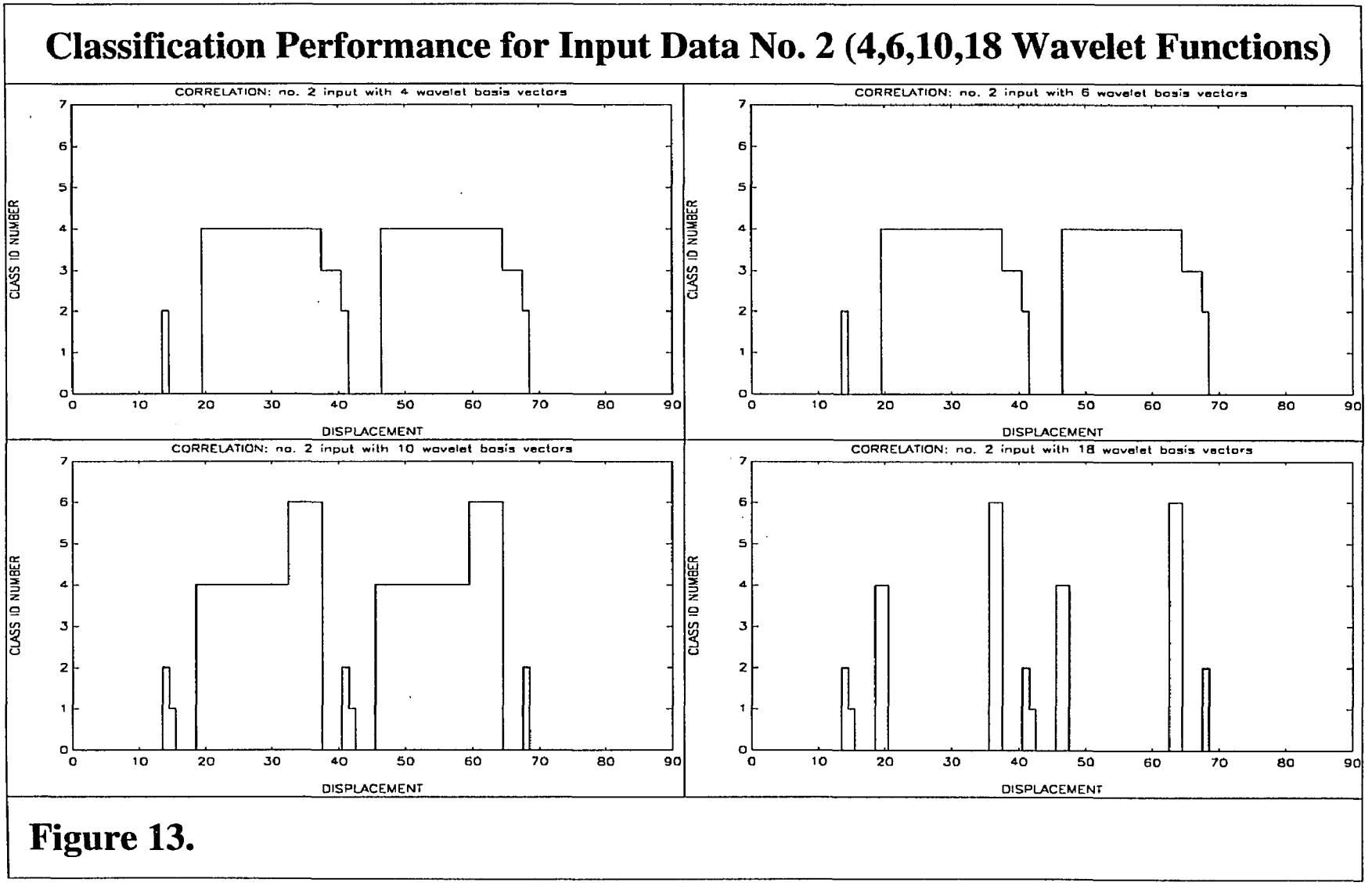
Classification Performance for Input Data No. 2 (4,6,10,18 Wavelet Functions)
CORRELATION: no. 2 input with 4 wavelet basis vectors
CORRELATION: no. 2 input with 6 wavelet basis vectors
CORRELATION: no. 2 input with 10 wavelet basis vectors
CORRELATION: no. 2 input with 18 wavelet basis vectors
CLASS ID NUMBER
DISPLACEMENT
0
1
2
3
4
5
6
7
0
10
20
30
40
50
60
70
80
90

Figure 13.

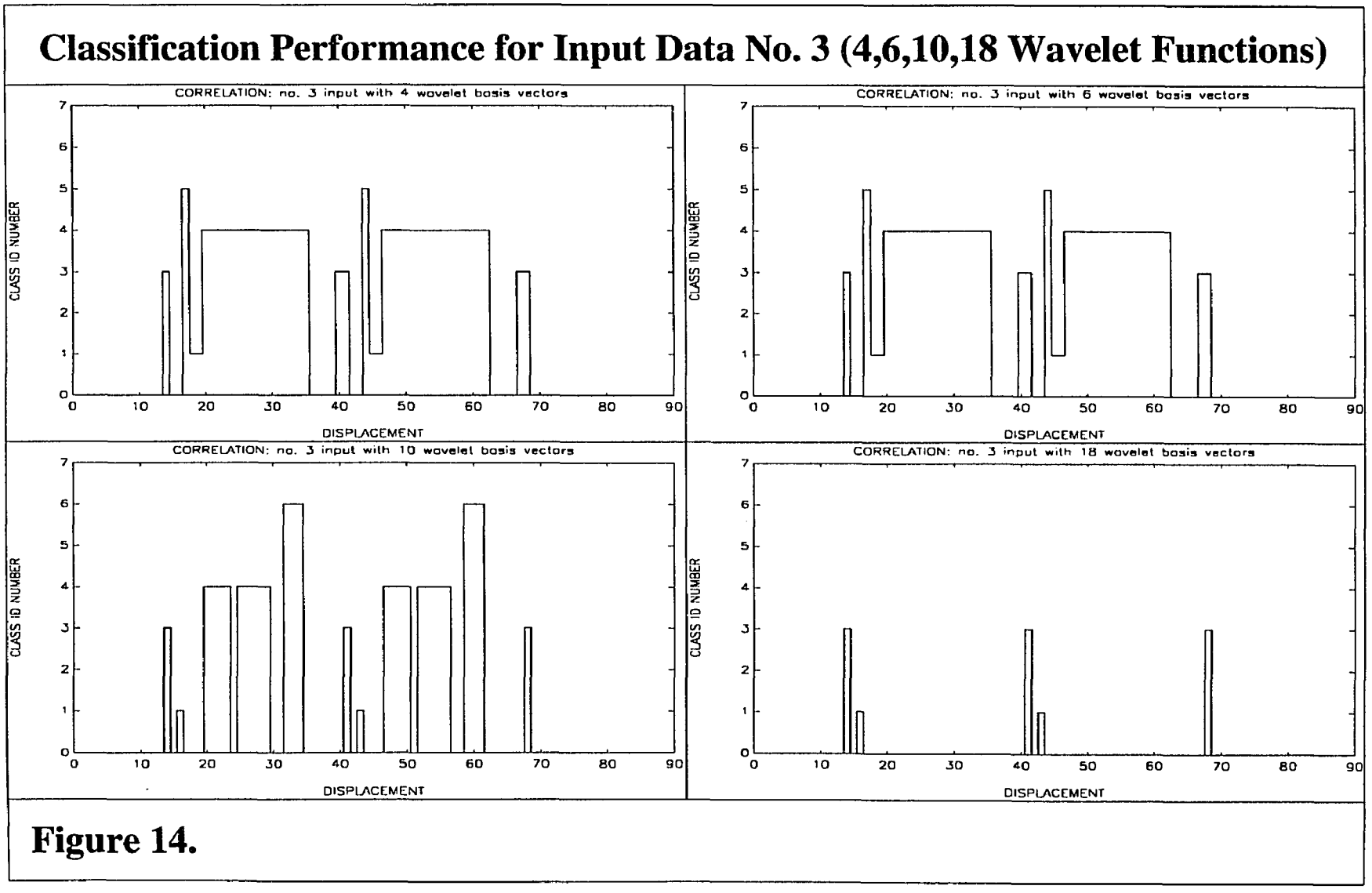
Classification Performance for Input Data No. 3 (4,6,10,18 Wavelet Functions)
CORRELATION: no. 3 input with 4 wavelet basis vectors
CORRELATION: no. 3 input with 6 wavelet basis vectors
CORRELATION: no. 3 input with 10 wavelet basis vectors
CORRELATION: no. 3 input with 18 wavelet basis vectors
CLASS ID NUMBER
DISPLACEMENT
0
10
20
30
40
50
60
70
80
90
1
2
3
4
5
6
7

Figure 14.

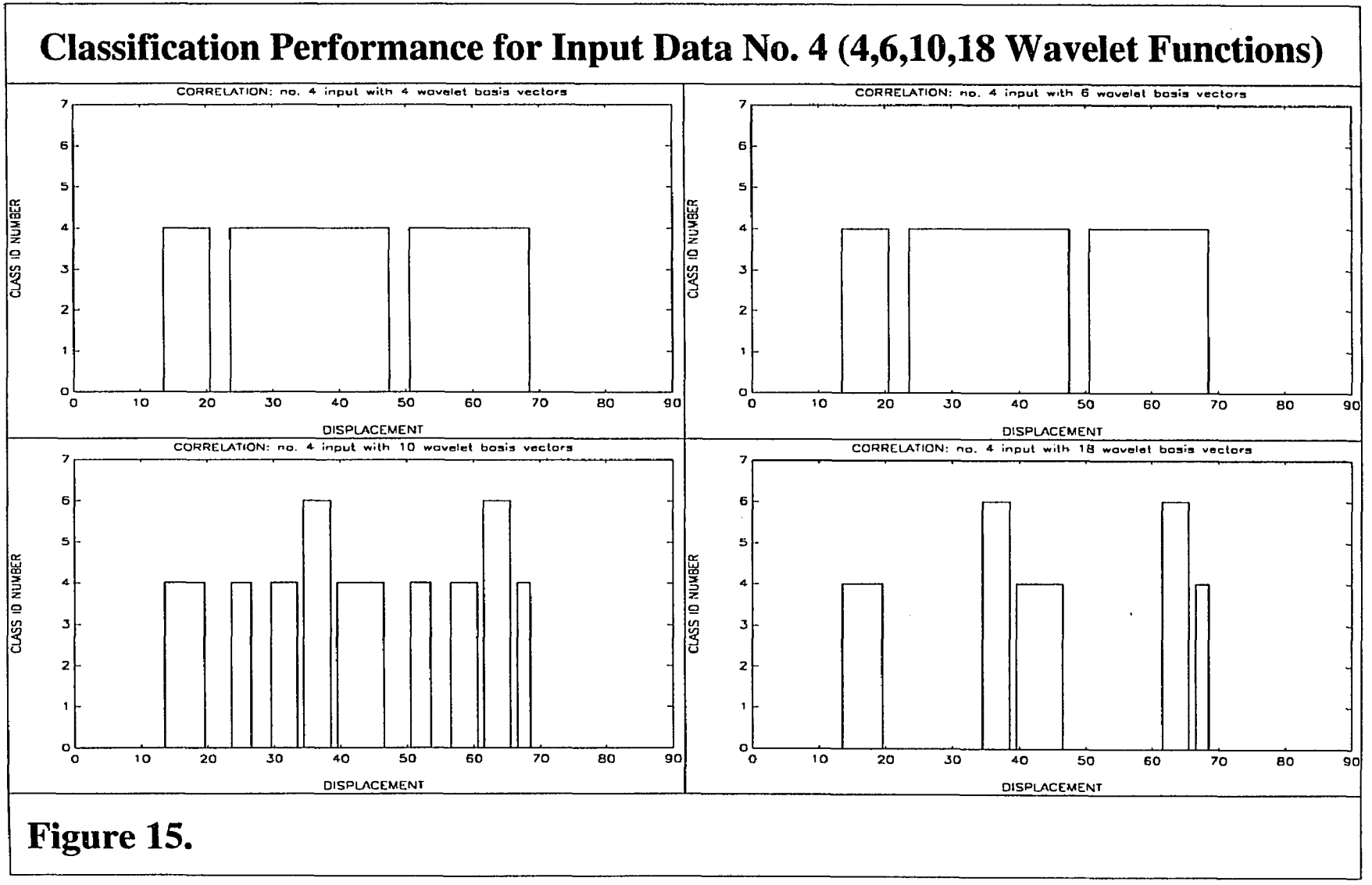
Classification Performance for Input Data No. 4 (4,6,10,18 Wavelet Functions)
CORRELATION: no. 4 input with 4 wavelet basis vectors
CORRELATION: no. 4 input with 6 wavelet basis vectors
CORRELATION: no. 4 input with 10 wavelet basis vectors
CORRELATION: no. 4 input with 18 wavelet basis vectors
CLASS ID NUMBER
DISPLACEMENT
0 10 20 30 40 50 60 70 80 90
0 1 2 3 4 5 6 7

Figure 15.

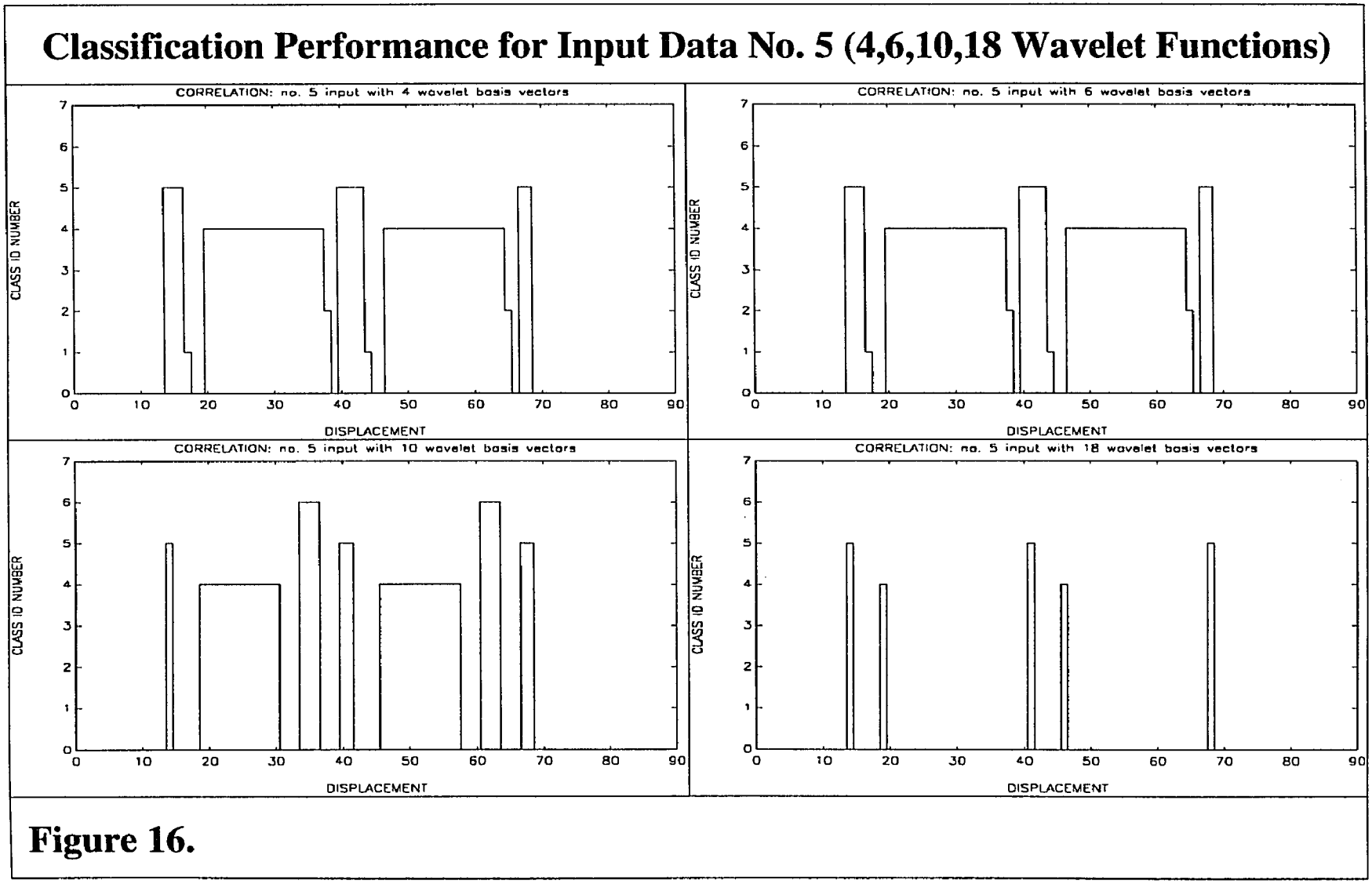
Classification Performance for Input Data No. 5 (4,6,10,18 Wavelet Functions)
CORRELATION: no. 5 input with 4 wavelet basis vectors
CORRELATION: no. 5 input with 6 wavelet basis vectors
CORRELATION: no. 5 input with 10 wavelet basis vectors
CORRELATION: no. 5 input with 18 wavelet basis vectors
CLASS ID NUMBER
DISPLACEMENT
0 10 20 30 40 50 60 70 80 90
0 1 2 3 4 5 6 7

Figure 16.

Classification Performance for Data Input No. 6 (4,6,10,18 Wavelet Functions)

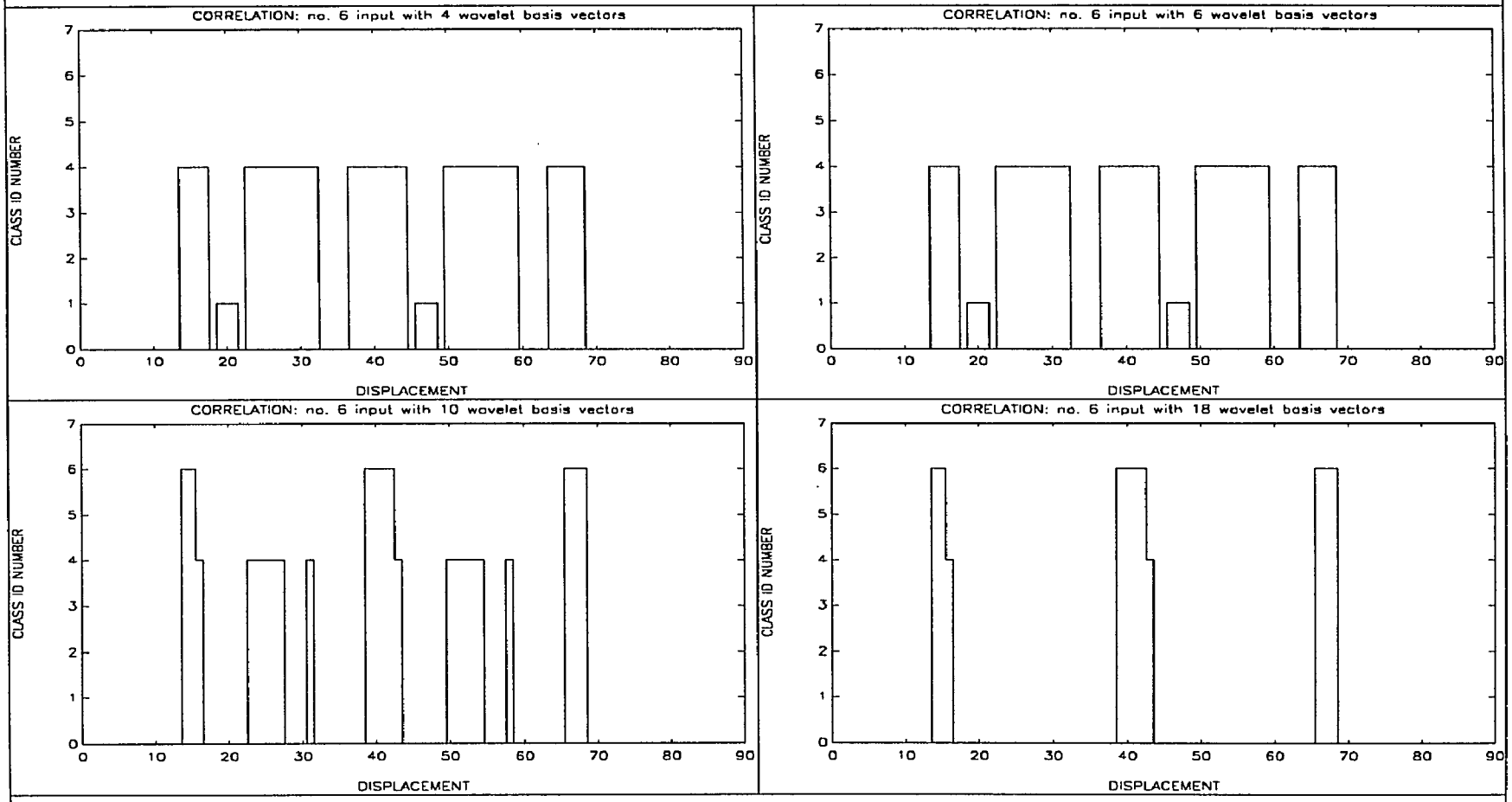

Figure 17.

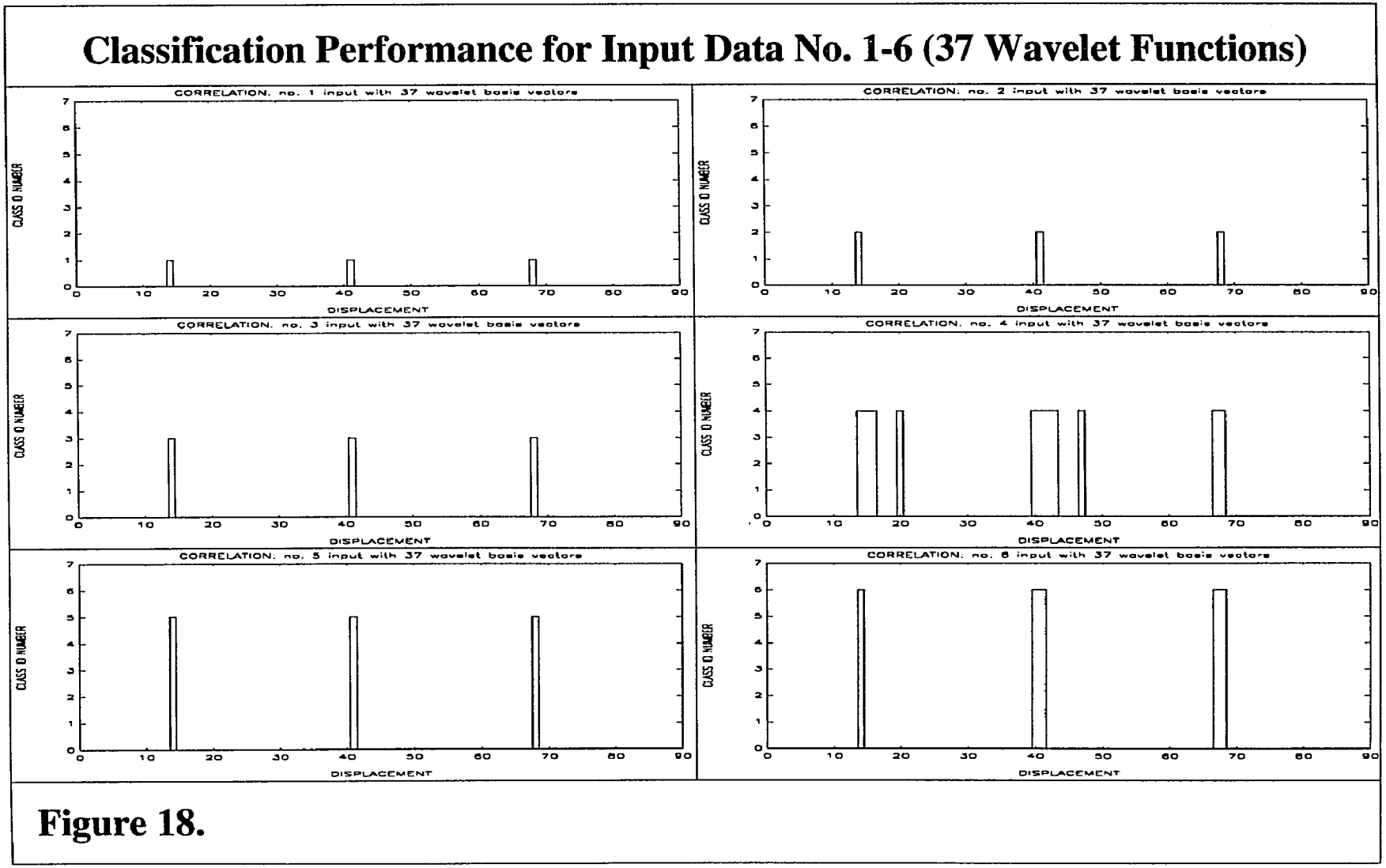
Classification Performance for Input Data No. 1-6 (37 Wavelet Functions)
CORRELATION. no. 1 input with 37 wavelet basis vectors
CORRELATION. no. 2 input with 37 wavelet basis vectors
CORRELATION. no. 3 input with 37 wavelet basis vectors
CORRELATION. no. 4 input with 37 wavelet basis vectors
CORRELATION. no. 5 input with 37 wavelet basis vectors
CORRELATION. no. 6 input with 37 wavelet basis vectors
CLASS ID NUMBER
DISPLACEMENT

Figure 18.

Hamming Distance Between Data Class No. 1-6 (6 Wavelet Functions)

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	0	1	7	10	8	8
Class 2	1	0	8	11	7	9
Class 3	7	8	0	15	9	13
Class 4	10	11	15	0	18	4
Class 5	8	7	9	18	0	16
Class 6	8	9	13	4	16	0

Sonlinh Phuvan, 17 December 1992

Table 4.

Hamming Distance Between Data Class No. 1-6 (18 Wavelet Functions)

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	0	1	3	2	1	3
Class 2	1	0	2	3	2	2
Class 3	3	2	0	3	2	2
Class 4	2	3	3	0	3	1
Class 5	1	2	2	3	0	4
Class 6	3	2	2	1	4	0

Sonlinh Phuvan, 17 December 1992, "wave02.cgm"

Table 5.

Hamming Distance Between Data Class No. 1-6 (37 Wavelet Functions)

	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
Class 1	0	2	7	3	7	2
Class 2	2	0	5	5	5	4
Class 3	7	5	0	8	2	7
Class 4	3	5	8	0	10	1
Class 5	7	5	2	10	0	9
Class 6	2	4	7	1	9	0

Sonlinh Phuvan, 17 December 1992, "wave04.cgm"

Table 6.

False Alarm Rate (4,6,10,18,37 Wavelet Functions)

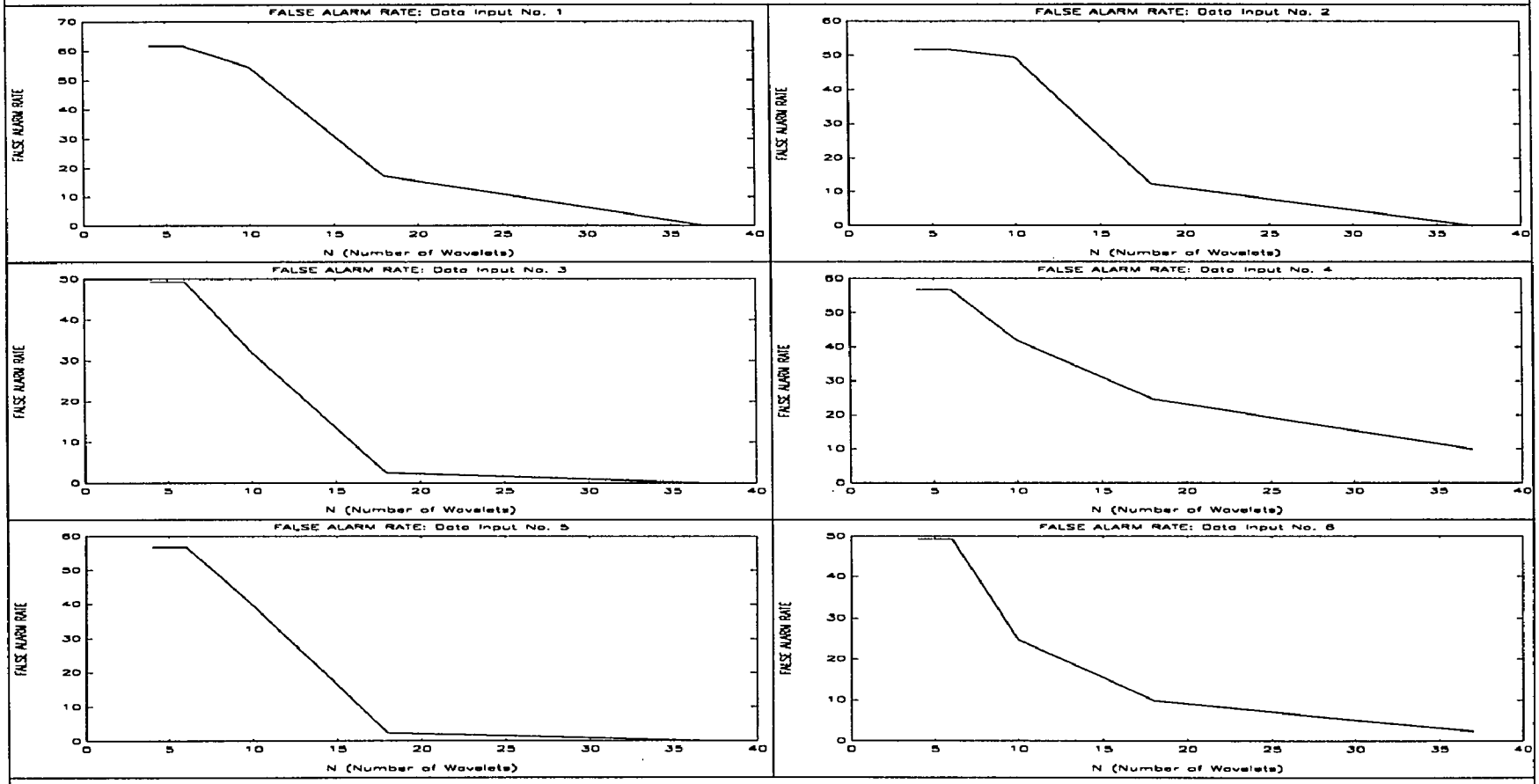

Figure 19.

3.3 Texture Characterization Using Polyfractals

3.3.1 Introduction

In the previous a method for uniquely identifying substructures g_j in a texture y_i was developed. That method, N-Wavelet coding allows scale and position invariant detection and classification with good performance with noise and distortion (Section 3.2).

In this section a method for obtaining a scale invariant measure of substructure g_j incidence is developed and demonstrated on two fractal textures. Two Cantor sets are produced and a polyfractal measure is obtained which can discriminate between the two sets, even if the two fractal set have identical fractal dimensions. This is because the polyfractal provide a vector of fractal measure of the subtextures which represents the difference between the two fractal textures.

3.3.2 Approach

Two cantor set basis (Fig. 20) will be used to produce two fractal substructures with a fractal dimensionality of 0.5 and 0.79 (Fig. 21,22). A set of classification feature will be extracted from the pseudo-Cantor sets (Fig. 23). This set of classification features will be used to develop wavelet functions (Fig.

24). This vector of wavelet functions can uniquely characterize the pseudo-Cantor sets by producing a unique classification pattern for each pseudo-Cantor sets (Table 7). A set of wavelet transforms will be performed on the pseudo-Cantor set using those wavelet functions or mother wavelets (Fig. 25-32). The wavelet transforms are thresholded, and the density of points above threshold is a measure of the number of times a particular feature is detected (Fig. 33,34). This density measure is plotted in a semilog format (Fig. 35-36) to obtain by linear regression a fractal measure estimate of each classification feature in each pseudo-Cantor sets (Table 8). The resultant vectors of the estimates of the fractal dimension for each basis feature will be used to obtain a **polyfractal** measure of the pseudo-Cantor sets. A polyfractal measure is a set of the fractal measures of each of the substructure.

A Cantor set , y , can be developed from iterating a Cantor basis structure, g. Let, g_0, be a basis Cantor structure, then the Cantor set can be obtained as follows,

$$
\begin{aligned}
g_0 &= [a_1 \quad a_2 \quad a_3 \quad \ldots \quad a_n] \\
g_1 &= [a_1 \times g_o \quad a_2 \times g_0 \quad a_3 \times g_0 \quad \ldots \quad a_n \times g_0] \\
&\ldots \\
y = g_{i+1} &= [a_1 \times g_i \quad a_2 \times g_i \quad a_3 \times g_i \quad \ldots \quad a_n \times g_i]
\end{aligned}
\tag{118}
$$

Then, y is the Cantor set produced from the basis cantor structure g_0. This basis cantor structure is equivalent to the substructure g_j of a texture y_i in the previous section.

Two Cantor basis structures (Fig. 20), or substructure g_j ,will be used to produce two pseudo-Cantor sets, y_i (Fig. 21,22).

3.3.3 Feature Extraction

An artificial neural network using a backpropagation learning paradigm is used to do feature extraction. Each neuron, k , in each neural layer is activated if the inner product of its associated connection weight, f_k, with the input, y_j , exceeds a threshold value.

$$z_{ik} = T_k[f_k \bullet y_i] \tag{119}$$

The inner product of two vectors is a measure of the similarity between the two vectors, if there is no similarities, the inner product is equal to zero, and the two vectors are said to be orthogonal. Thus, $\langle f_k \mid y_i \rangle$ is a point by point correlation with fixed displacement of f_k and the input y_i . f_k are then the set of features that the neural uses to perform classification.

In order to use a neural net to do feature extraction, the two Cantor basis structures are used as input classes. The artificial neural net is fully connected and has four input nodes, and two hidden layers with 4, 3 neural nodes respectively and 2 neural nodes in the output. The artificial neural net is then run until it converges, i.e. it discriminates between the two Cantor basis structures. Each weight vector associated with each neuron in the first layer can then be considered to be a feature, f_k, that the neural net has selected to be significant for feature extraction (Fig. 23). The classification pattern required to classify each Cantor set is given by a set of 0's and 1's reflecting the absence or presence of a particular feature (Table 7).

3.3.4 Wavelet Design

Each of the features f_k extracted by the artificial neural net is used to produce wavelet functions w_k. A wavelet function has a zero moment, usually a zero mean, and finite support, the value of wavelet function decays at least geometrically with extent.

To ensure that each wavelet function, w_k, has zero mean, the mean value of the extracted feature is substracted from itself,

$$v_k = f_k - \bar{f}_k \tag{120}$$

then, v_k , is scaled to ensure that its inner product with the Cantor basis is equal to that of the inner product of extracted feature and the Cantor set basis,

$$w_k = \left\{ \min_{j, e_k > 0} \frac{e_{kj}}{g_j \bullet v_k} \right\} v_k \tag{121}$$

where $e_{kj} = g_j \bullet f_k - d_k$, and

where d_k is the bias term at neural node k of the input layer of the artificial neural net used to do feature extraction. In Fig. 23, the bias terms for f_k, where k = 1-4, are respectively, -0.5520, -0.7750, -0.9790 and -0.4840.

Once the wavelet functions are developed (Fig. 24), the wavelet classification pattern t_{jk} used to classify the input can be obtained by the threshold of the inner product of the Cantor basis structure g_j and the wavelet functions w_k.

$$t_{jk} = (g_j \bullet w_k) > 0 \tag{122}$$

3.3.5 Fractal Measure

Each of the wavelet function can be considered to be a mother wavelet, a Wavelet transform for each of the Cantor set is obtained using all mother wavelet functions.

$$u = \mathrm{W}[y, w] \tag{123}$$

$$u_{ij}(a,b) = \frac{1}{a}\sum_{h}^{n} y(i,h)w\left(j,\frac{h-b}{a}\right) \tag{124}$$

where,

a = scale

b = displacement

Thus, one can think of the wavelet transform as a correlation with respect to scale, and the amplitude of the wavelet transform is the correlation amplitude with displacement b and scale a.

For each pseudo-Cantor set four scales are used for each mother wavelet then subsequently thresholded for feature detection (Fig. 25-32).

For each pattern class the number of detection, N_a, is measured and the fractal dimension, D, corresponding to that pattern class or basis substructure is found in the limit of descending scale a with respect to N_a.

$$N_a \sim a^{-D} \tag{125}$$

$$D \sim \lim_{a\to 0} \frac{\log(N_a)}{\log(1/a)} \tag{126}$$

The number of detection, N_a, versus the scale of the wavelet function used during the detection is plotted for each pseudo-Cantor sets (Fig. 33-34). A semi-log plot is obtained (Fig. 35-36) and a linear regression is used to obtain an estimate of the slope, or of fractal dimension, D, as obtained for each wavelet functions (Table 8).

3.3.6 Conclusions

As can be seen from Table 8, each Wavelet function provide a significantly different dimensional estimate for each pseudo-Cantor set. The set of fractal dimension estimates for each pseudo-Cantor set obtained from the Wavelet transforms is called a **polyfractal** measure to differentiate from other techniques and produces a unique characterization of fractal textures.

The polyfractal provide a statistical measure of the incidence of the difference between textures, and thus a unique characterization can be obtained, for any texture.

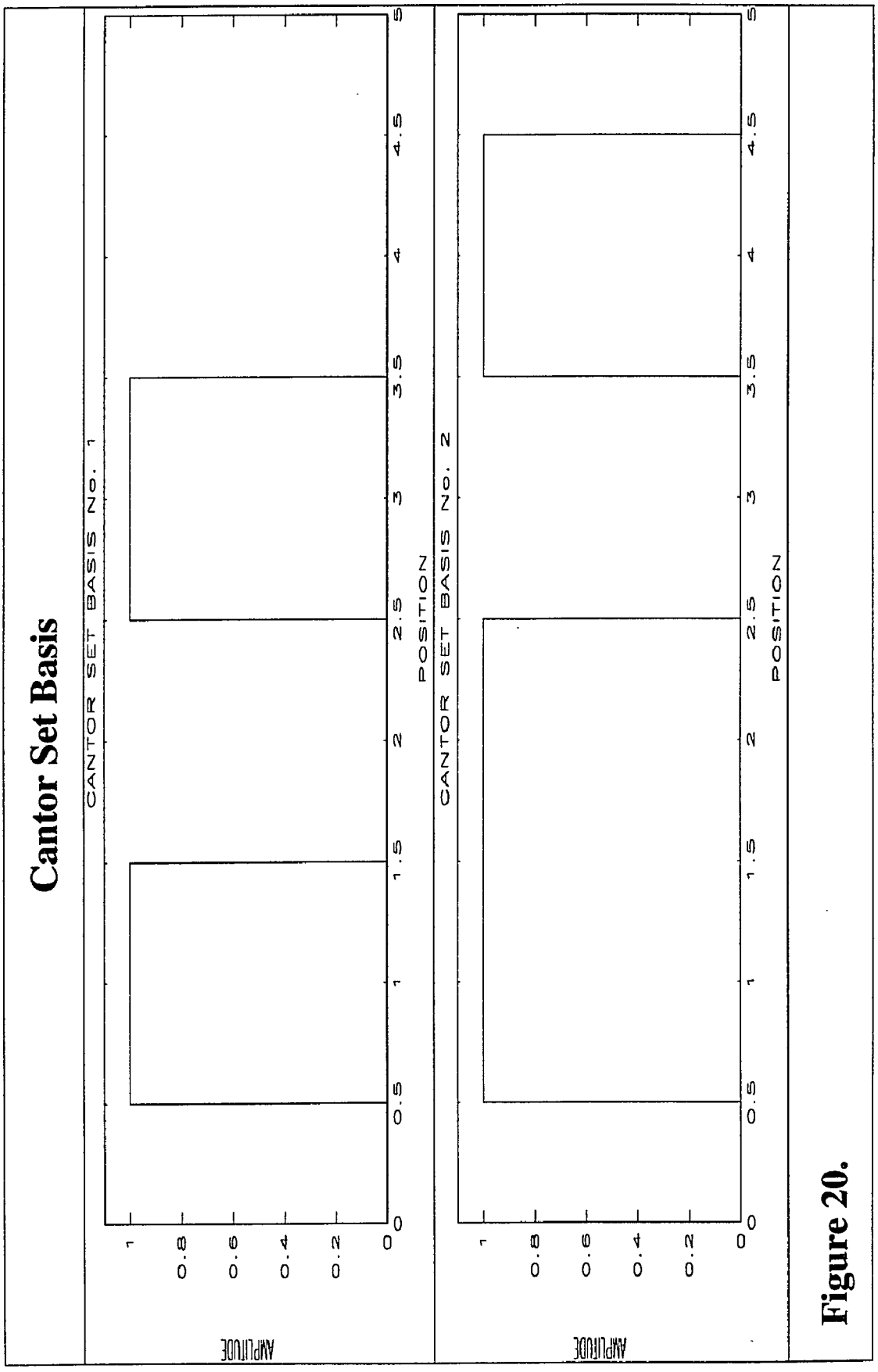
Cantor Set Basis
CANTOR SET BASIS No. 1
CANTOR SET BASIS No. 2
AMPLITUDE
POSITION
0
0.2
0.4
0.6
0.8
1
0.5
1.5
2
2.5
3
3.5
4
4.5
5

Figure 20.

Cantor Set 1

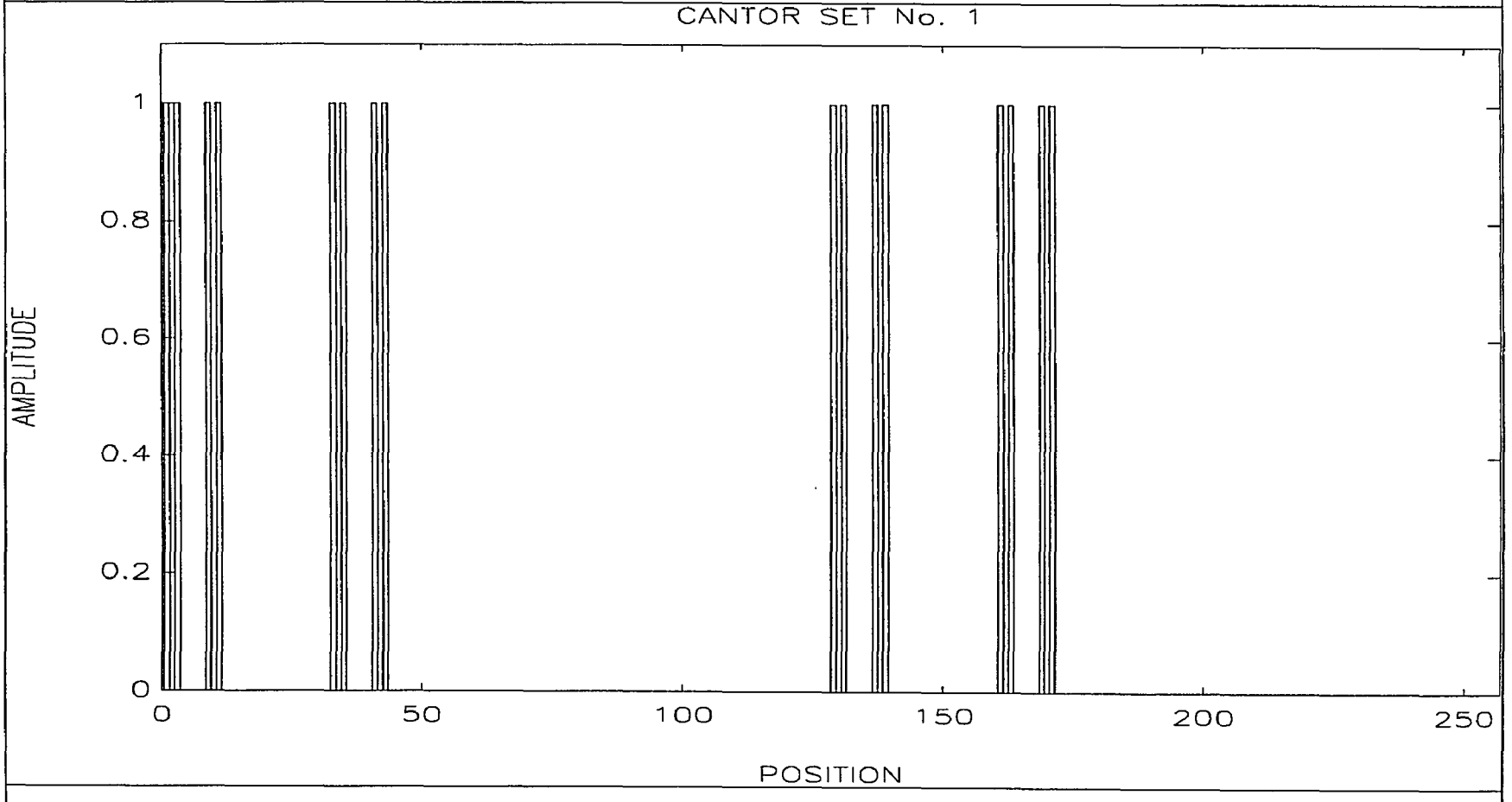

Figure 21.

Cantor Set 2

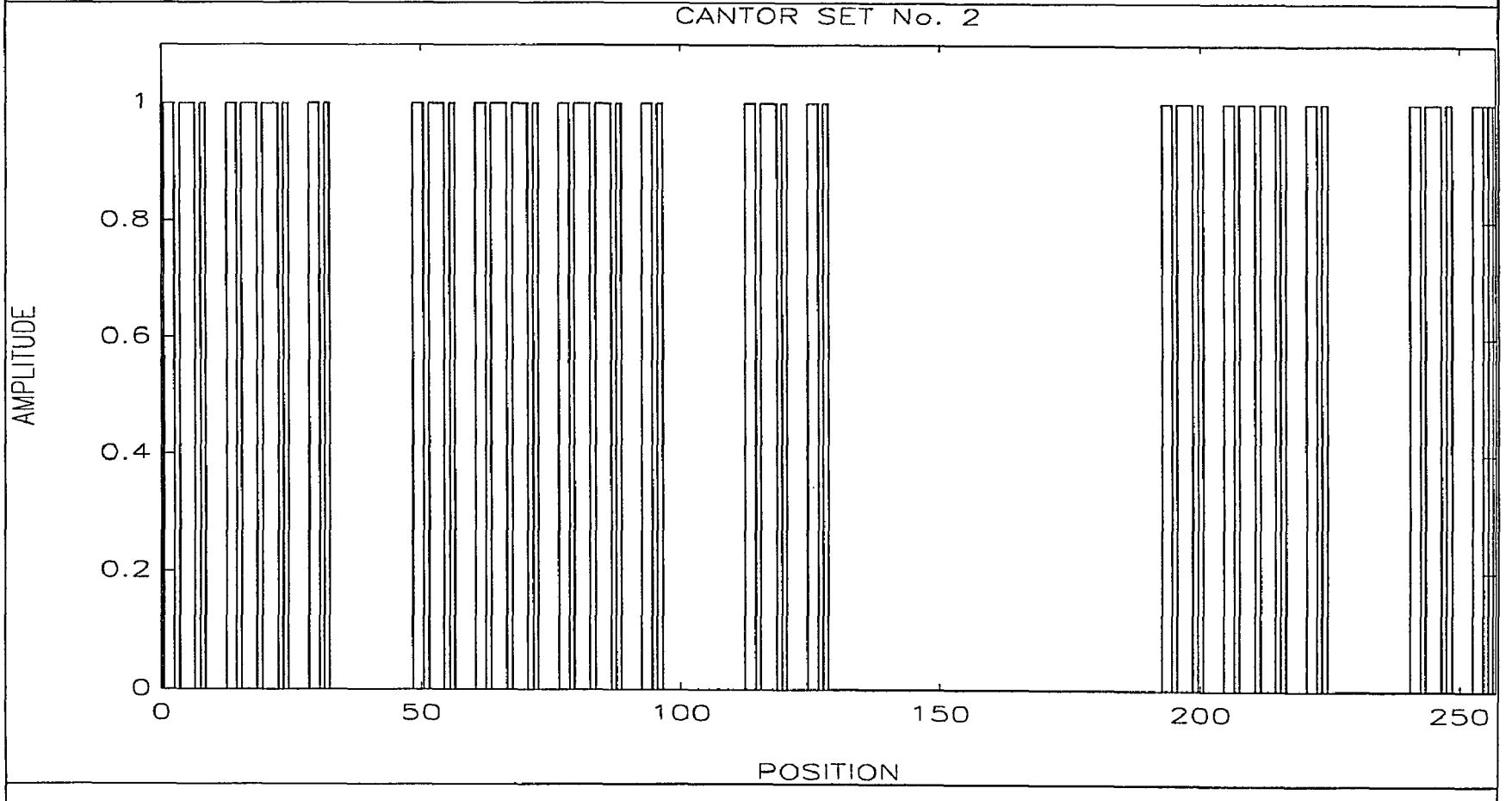

Figure 22.

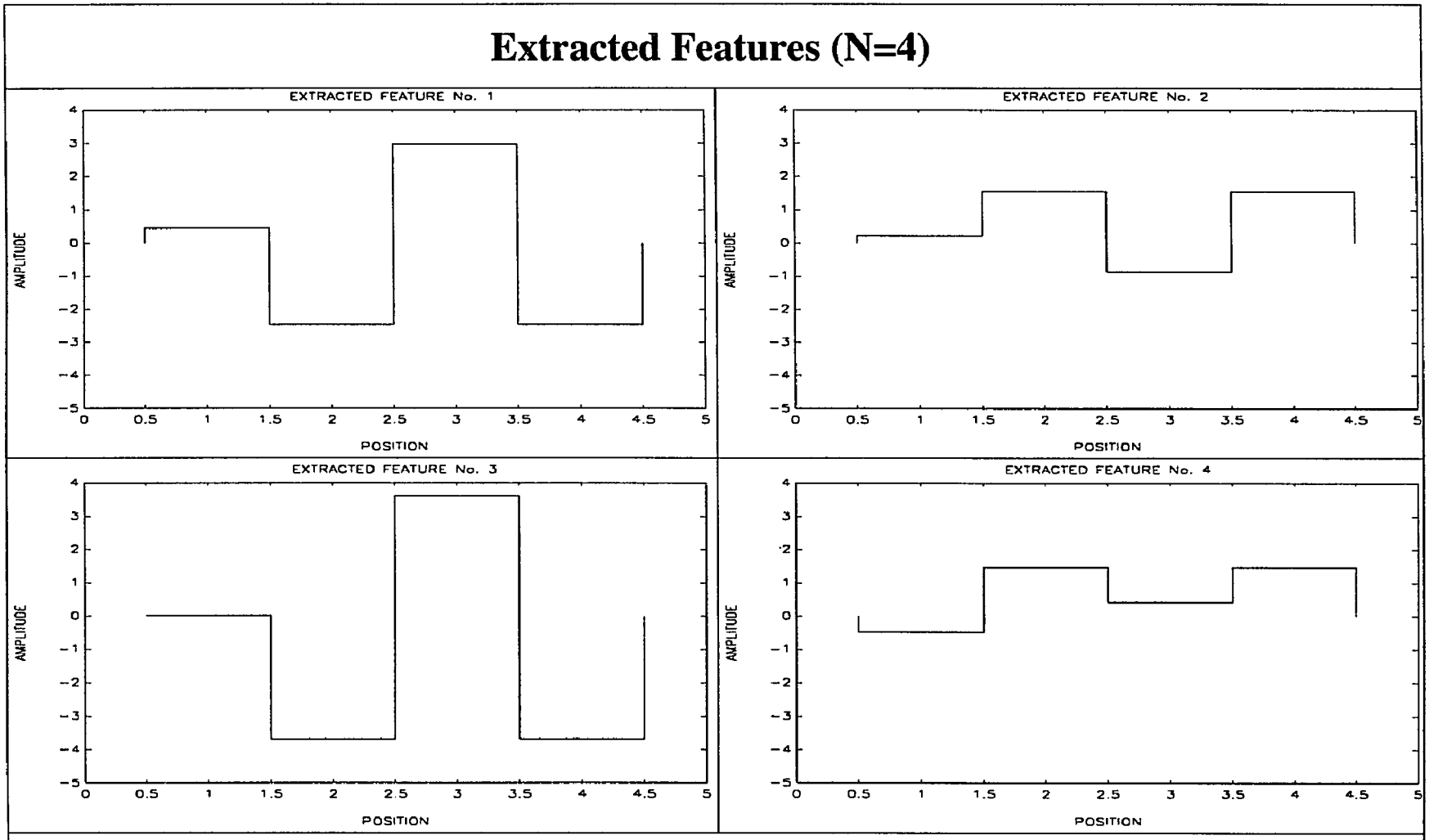
Extracted Features (N=4)
EXTRACTED FEATURE No. 1
EXTRACTED FEATURE No. 2
EXTRACTED FEATURE No. 3
EXTRACTED FEATURE No. 4
AMPLITUDE
POSITION

Figure 23.

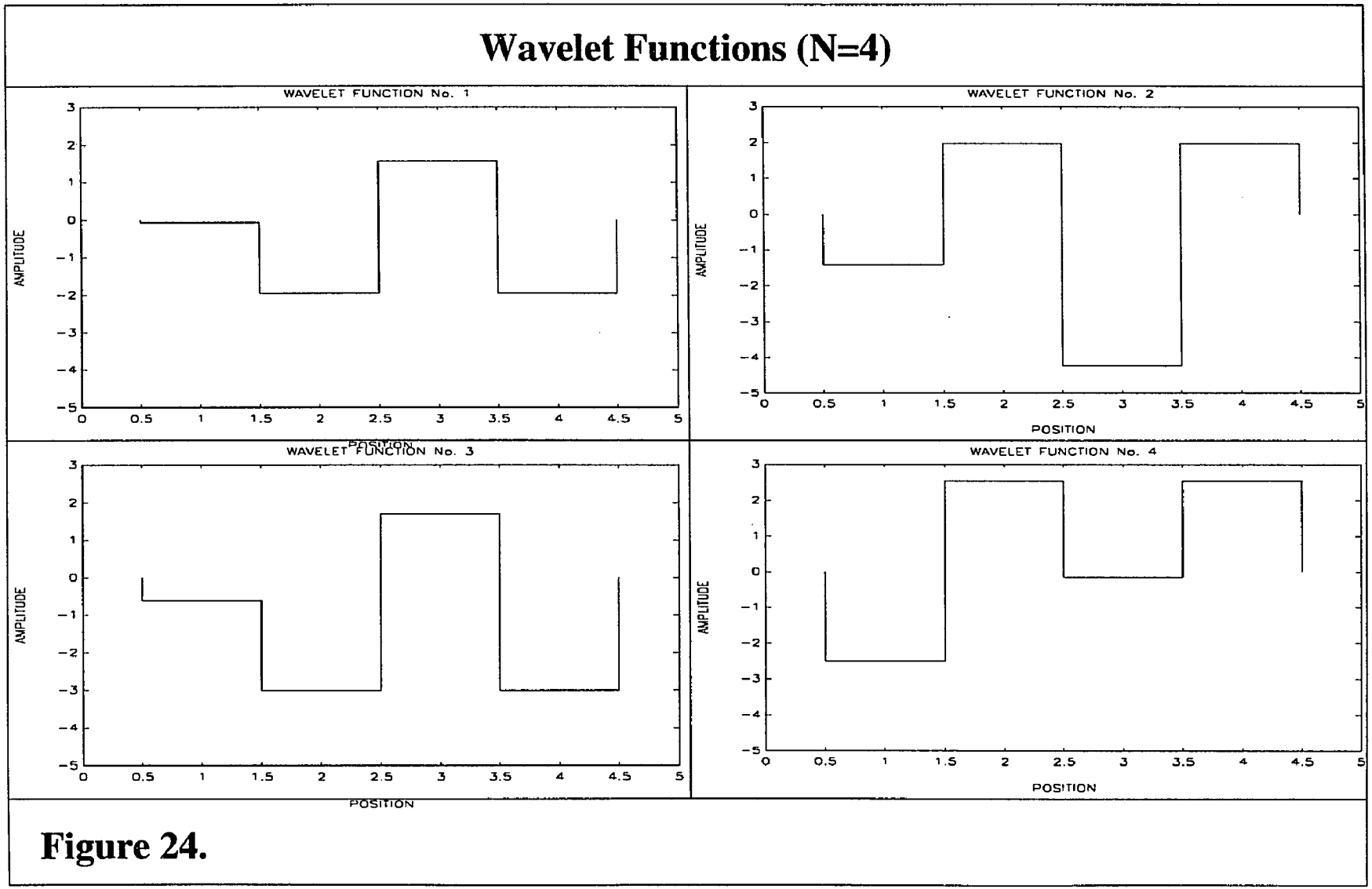
Wavelet Functions (N=4)
WAVELET FUNCTION No. 1
WAVELET FUNCTION No. 2
WAVELET FUNCTION No. 3
WAVELET FUNCTION No. 4
AMPLITUDE
POSITION

Figure 24.

Cantor Set Classification Pattern

CANTOR SET CLASSIFICATION PATTERN

	WAVELET 1	WAVELET 2	WAVELET 3	WAVELET 4
CANTOR SET No. 1	1	0	1	0
CANTOR SET No. 2	0	1	0	1

Sonlinh Phuvan. 14 January 1993

Table 7.

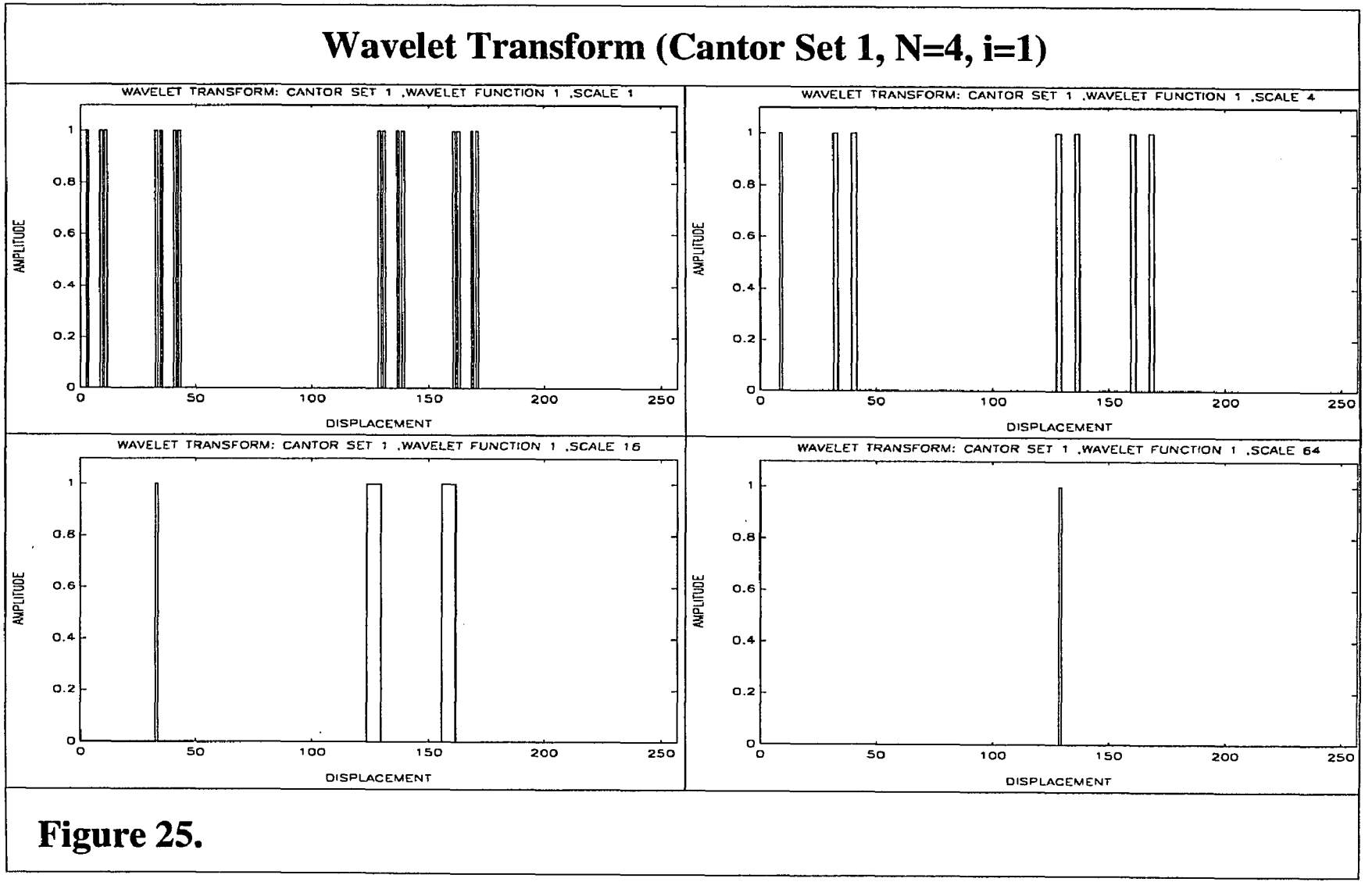
Wavelet Transform (Cantor Set 1, N=4, i=1)
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 1 ,SCALE 1
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 1 ,SCALE 4
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 1 ,SCALE 16
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 1 ,SCALE 64
AMPLITUDE
DISPLACEMENT
0
0.2
0.4
0.6
0.8
1
50
100
150
200
250

Figure 25.

Wavelet Transform (Cantor Set 1, N=4, i=2)

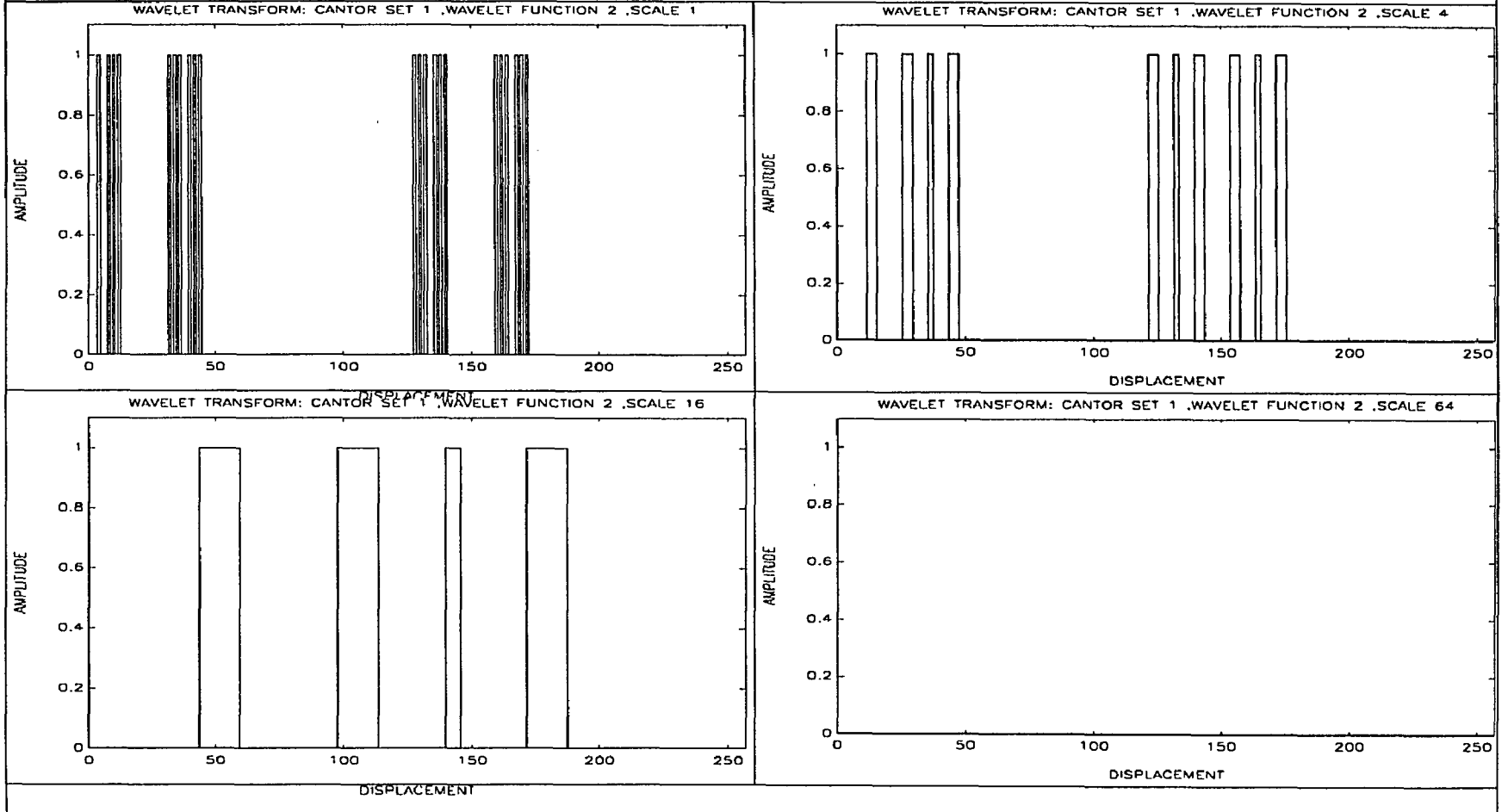

Figure 26.

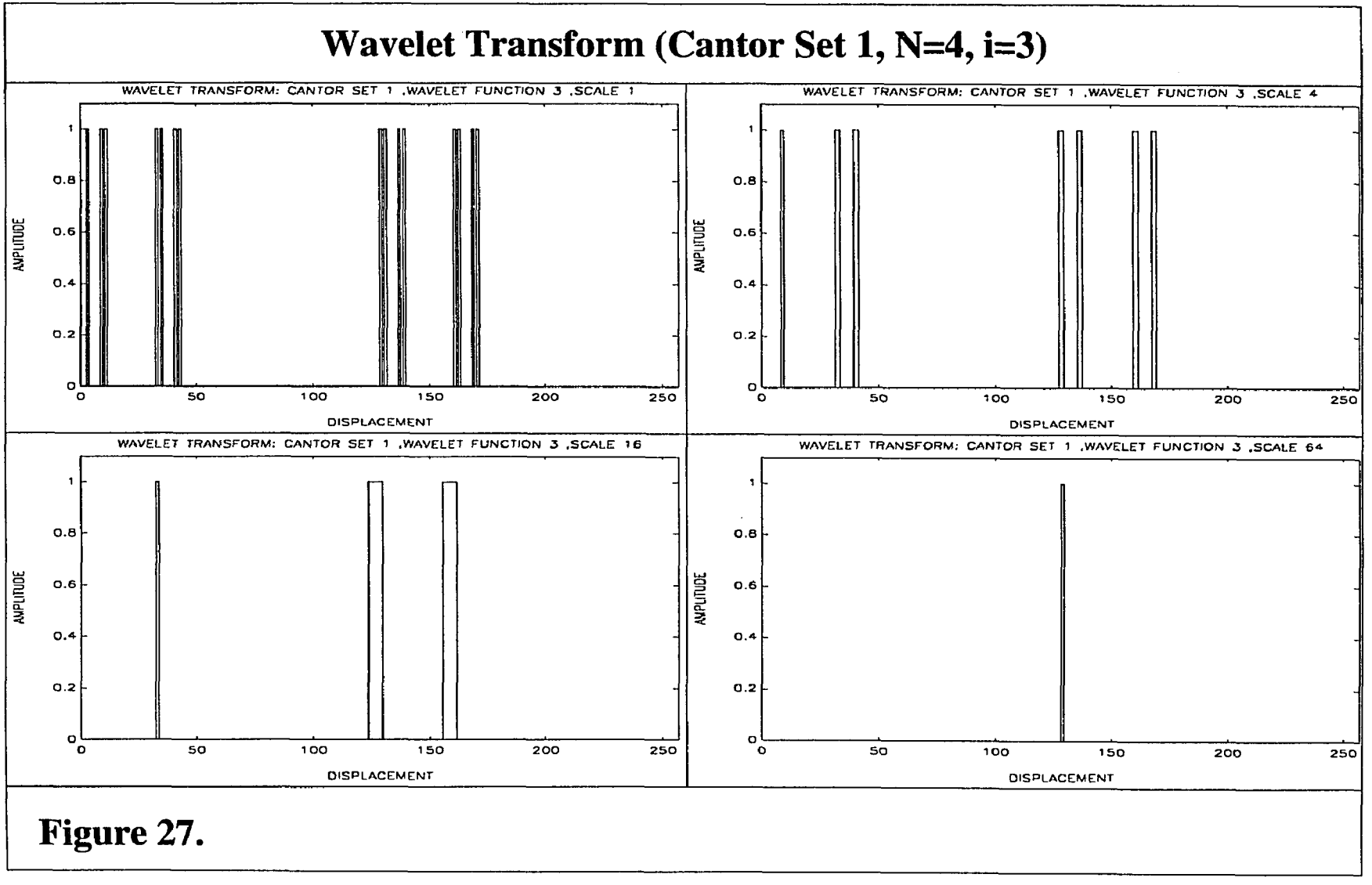
Wavelet Transform (Cantor Set 1, N=4, i=3)
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 3 ,SCALE 1
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 3 ,SCALE 4
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 3 ,SCALE 16
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 3 ,SCALE 64
AMPLITUDE
DISPLACEMENT
0
0.2
0.4
0.6
0.8
1
0
50
100
150
200
250

Figure 27.

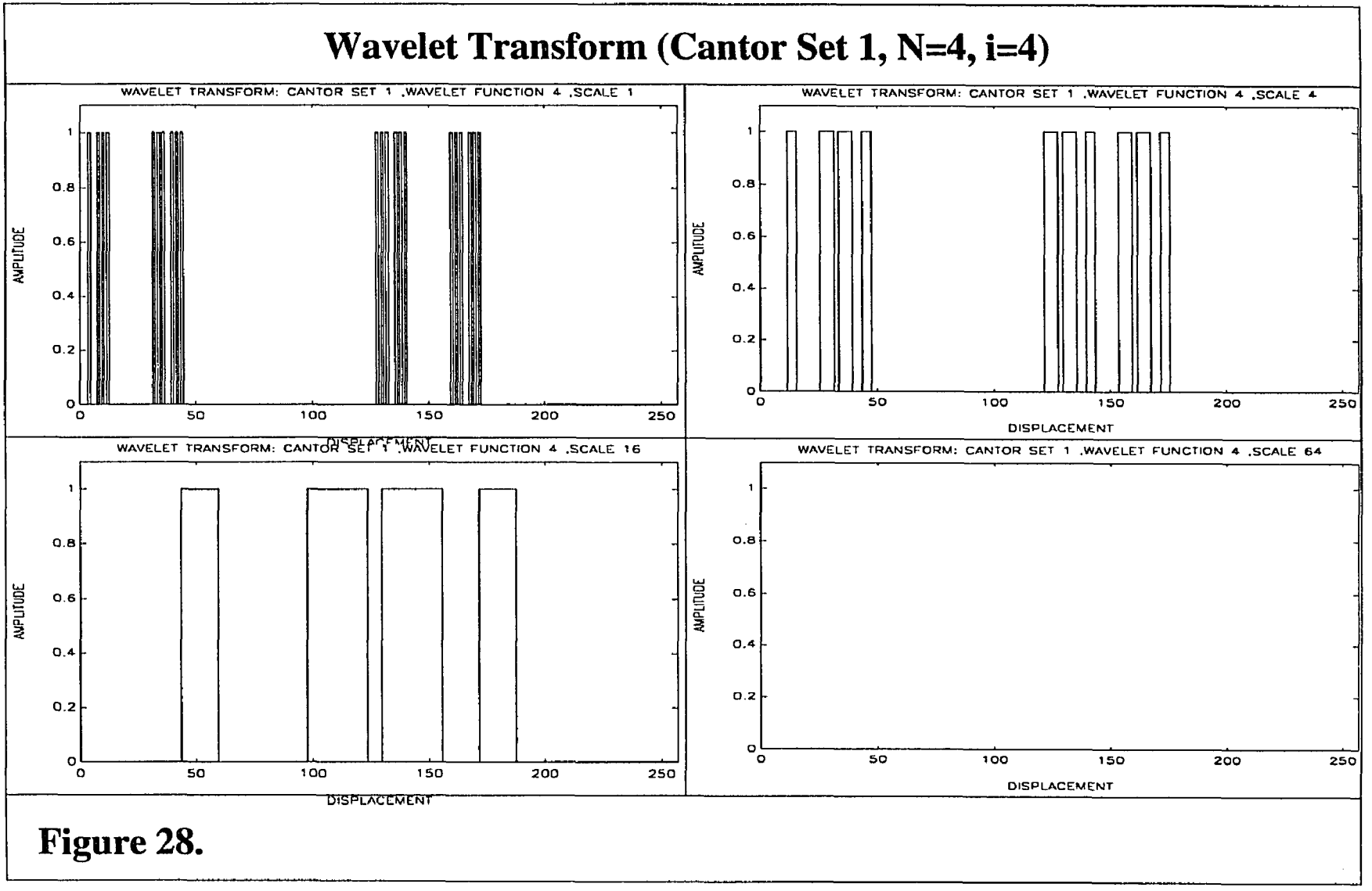
Wavelet Transform (Cantor Set 1, N=4, i=4)
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 4 ,SCALE 1
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 4 ,SCALE 4
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 4 ,SCALE 16
WAVELET TRANSFORM: CANTOR SET 1 ,WAVELET FUNCTION 4 ,SCALE 64
AMPLITUDE
DISPLACEMENT
0
0.2
0.4
0.6
0.8
1
50
100
150
200
250

Figure 28.

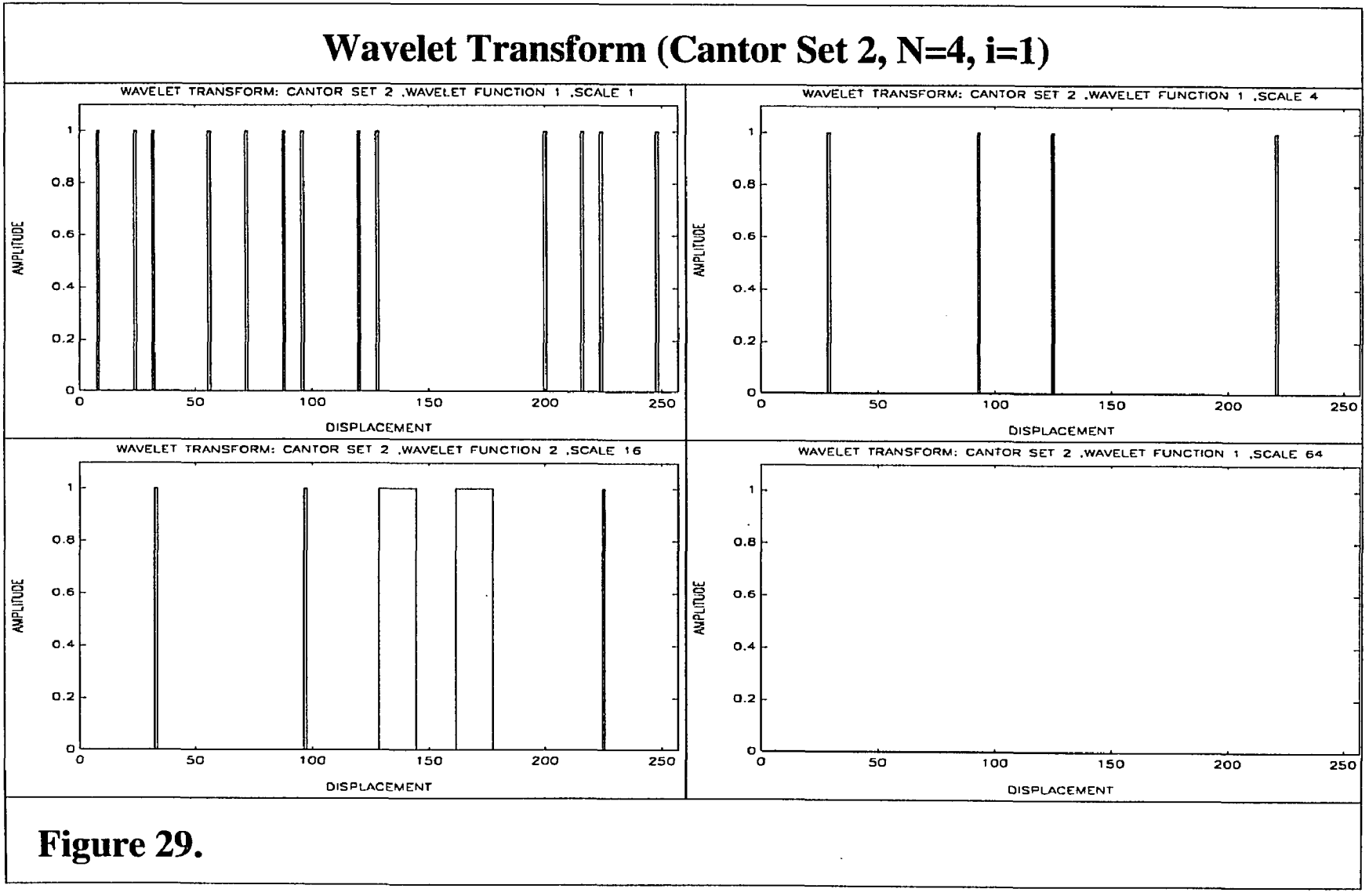
Wavelet Transform (Cantor Set 2, N=4, i=1)
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 1 ,SCALE 1
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 1 ,SCALE 4
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 2 ,SCALE 16
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 1 ,SCALE 64
AMPLITUDE
DISPLACEMENT
0
0.2
0.4
0.6
0.8
1
50
100
150
200
250

Figure 29.

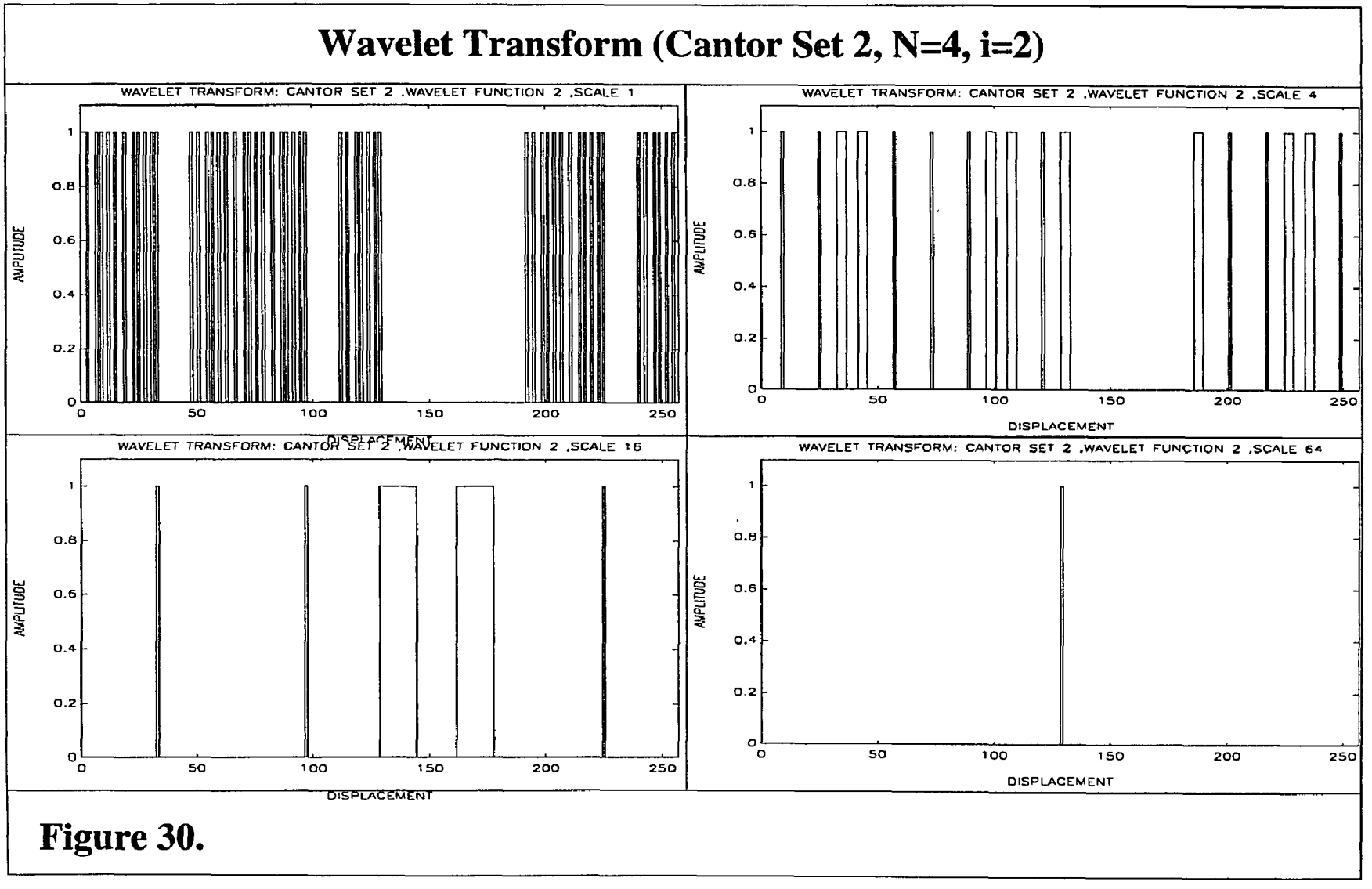
Wavelet Transform (Cantor Set 2, N=4, i=2)
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 2 ,SCALE 1
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 2 ,SCALE 4
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 2 ,SCALE 16
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 2 ,SCALE 64
AMPLITUDE
DISPLACEMENT
0
0.2
0.4
0.6
0.8
1
50
100
150
200
250

Figure 30.

Wavelet Transform (Cantor Set 2, N=4, i=3)

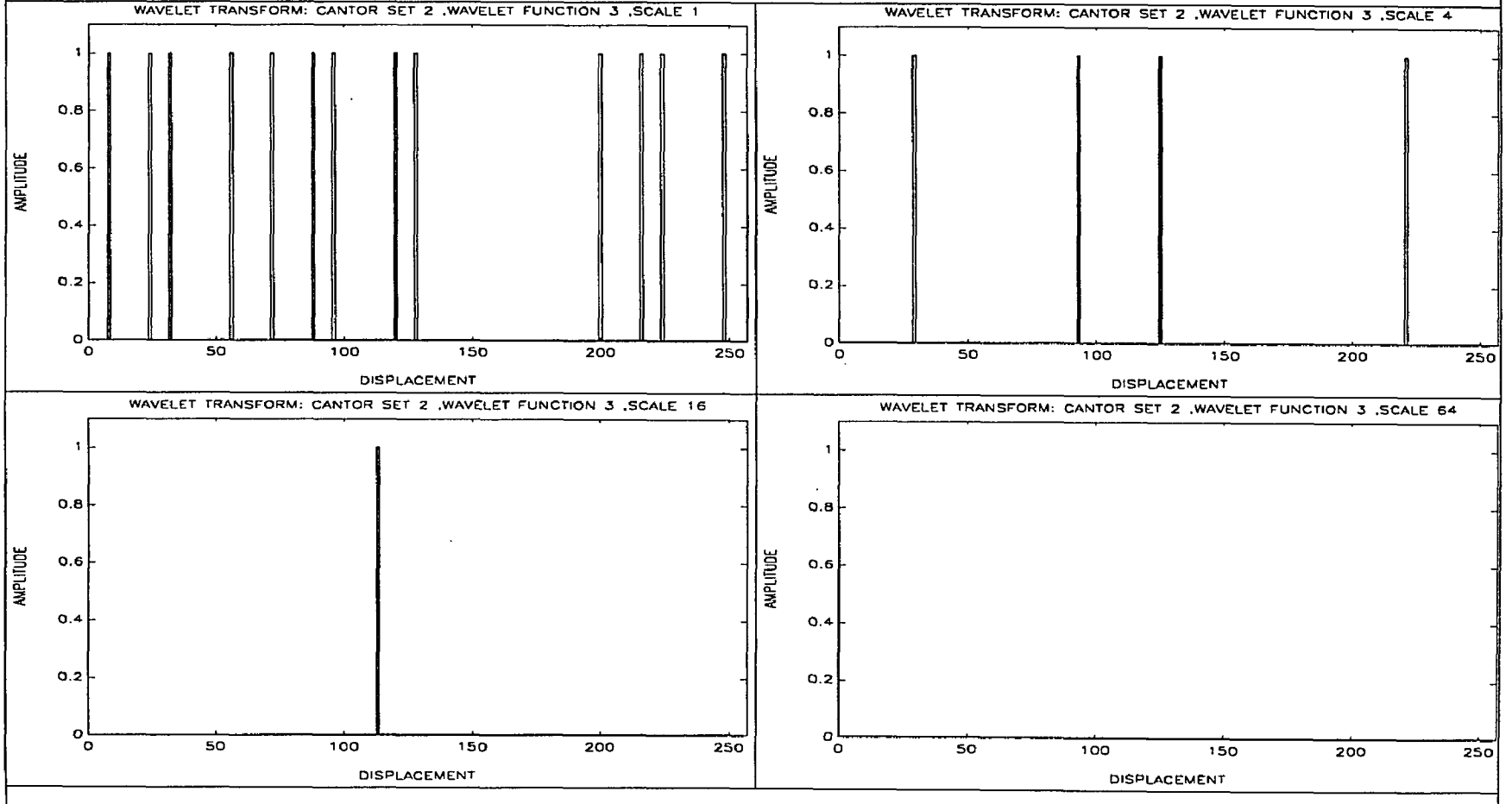

Figure 31.

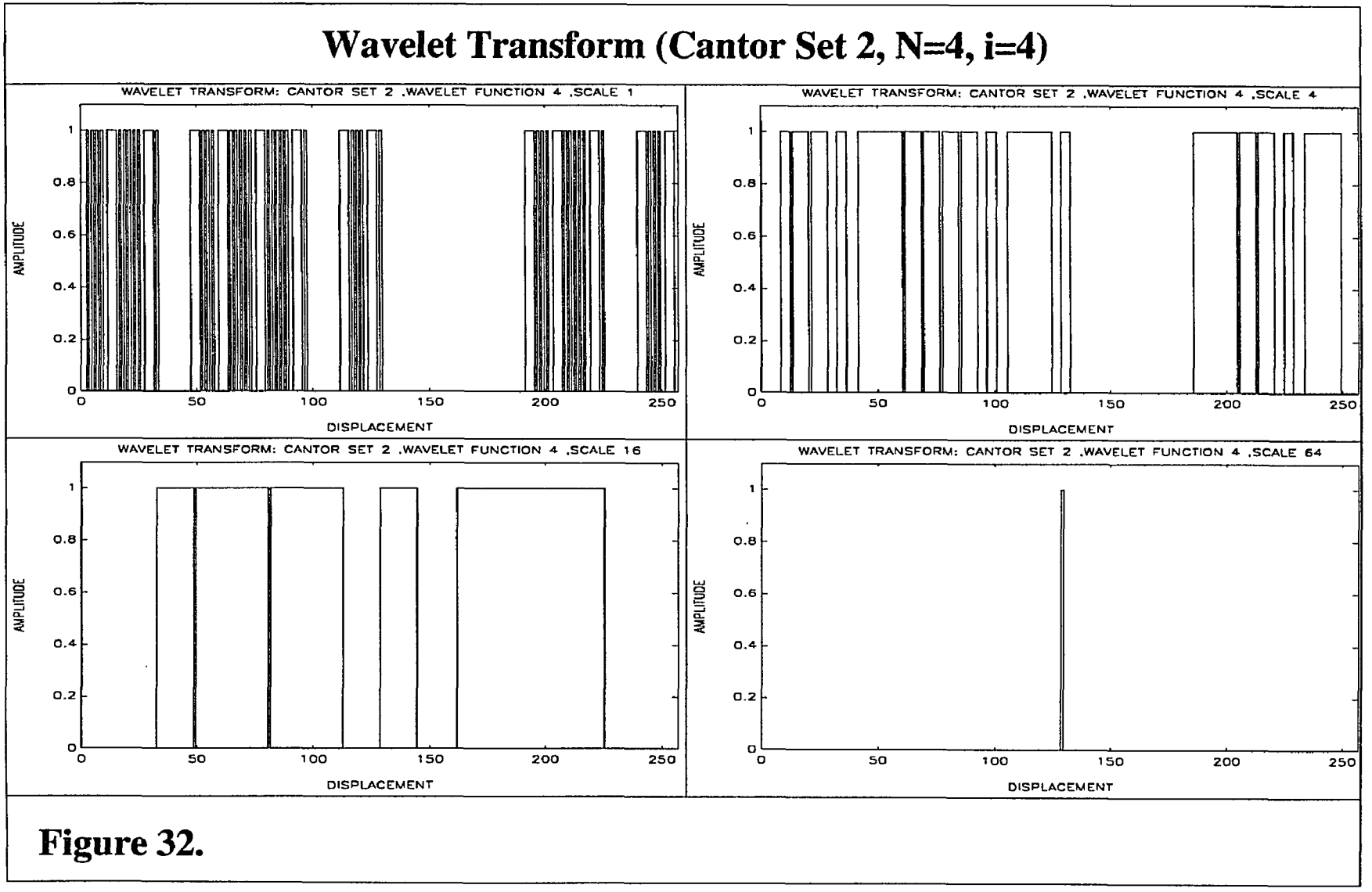
Wavelet Transform (Cantor Set 2, N=4, i=4)
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 4 ,SCALE 1
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 4 ,SCALE 4
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 4 ,SCALE 16
WAVELET TRANSFORM: CANTOR SET 2 ,WAVELET FUNCTION 4 ,SCALE 64
AMPLITUDE
DISPLACEMENT
0
0.2
0.4
0.6
0.8
1
50
100
150
200
250

Figure 32.

Feature Density Measure (Cantor Set 1, N=4)

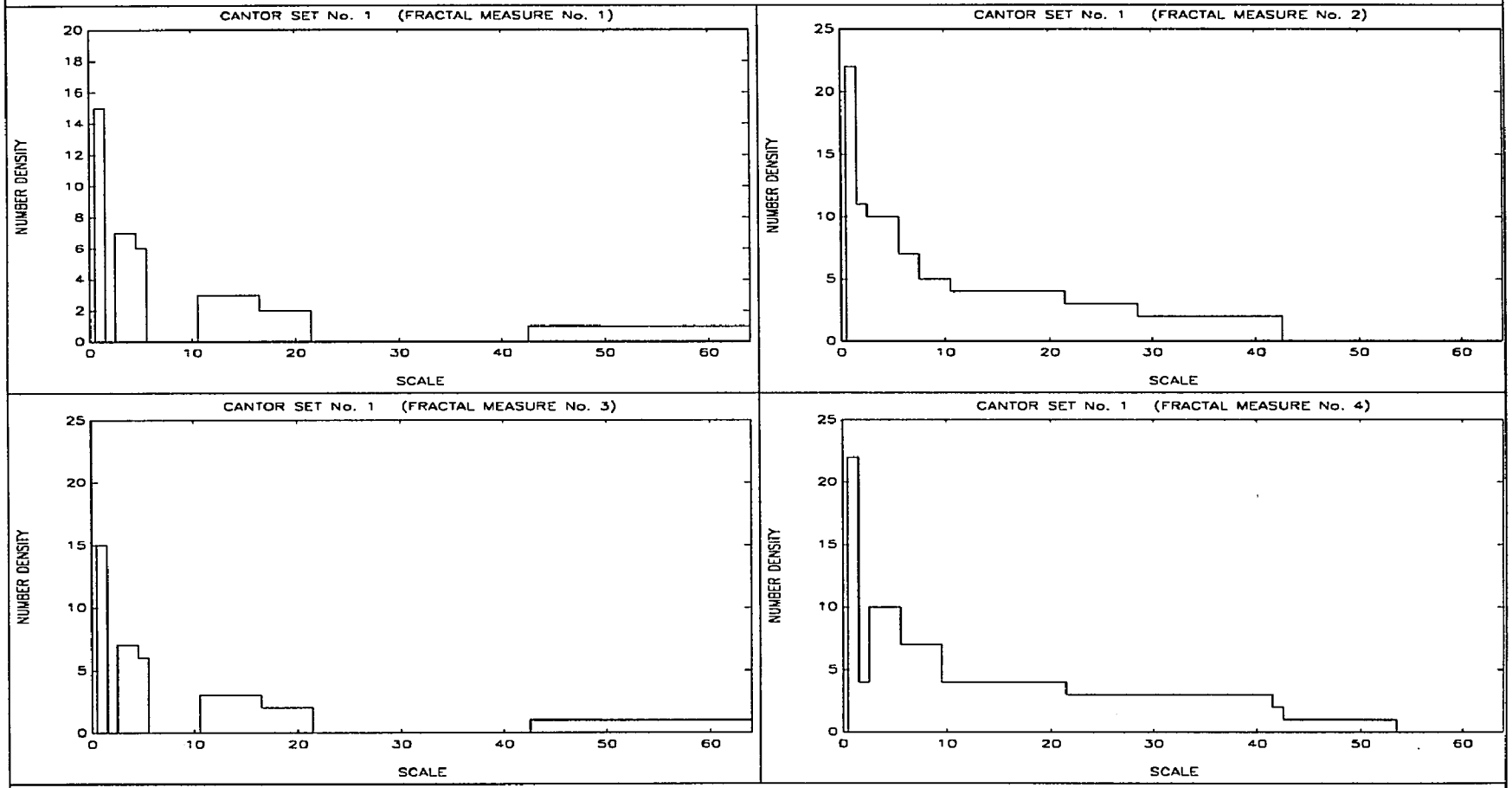

Figure 33.

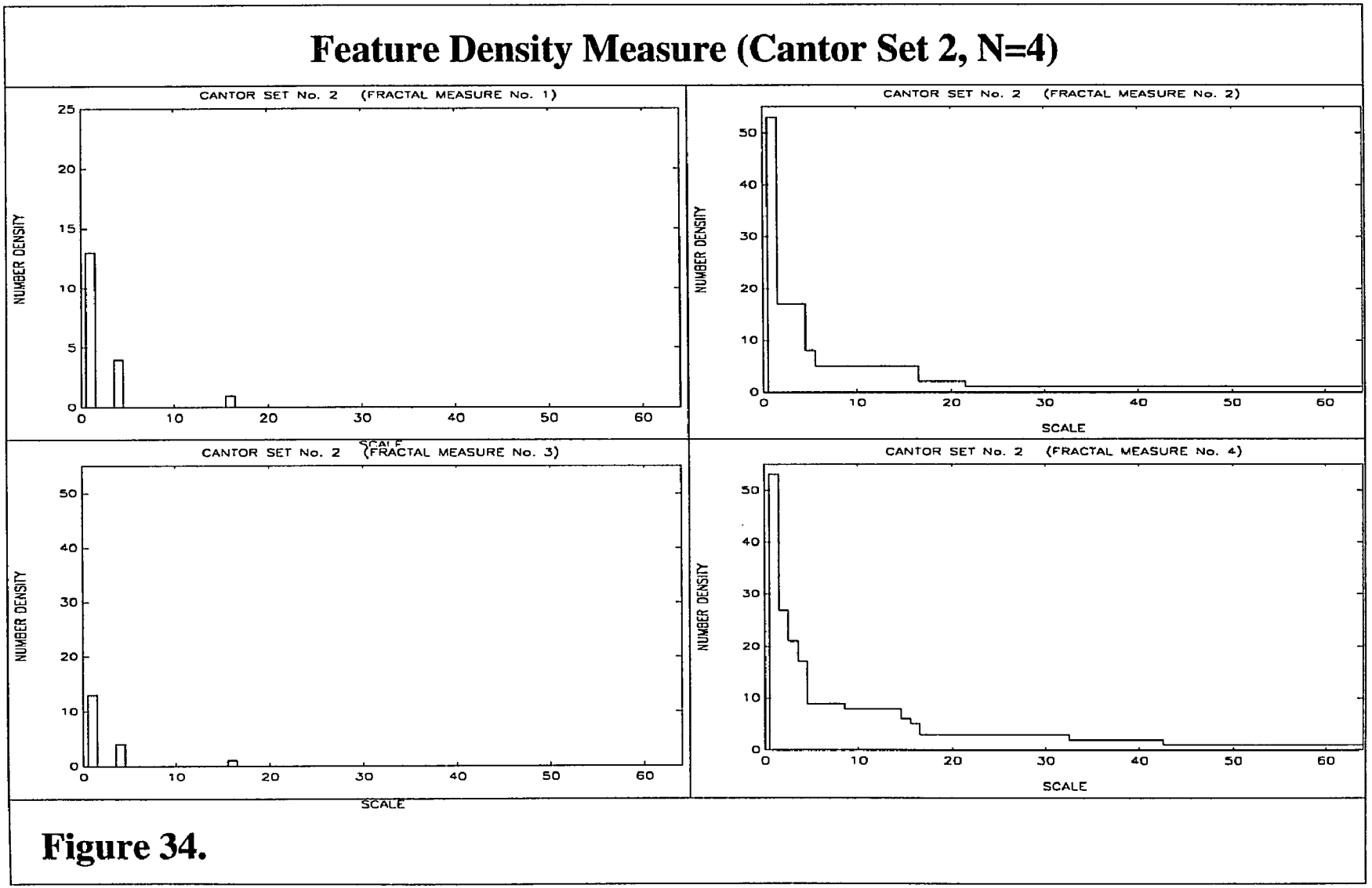
Feature Density Measure (Cantor Set 2, N=4)
CANTOR SET No. 2 (FRACTAL MEASURE No. 1)
NUMBER DENSITY
SCALE
CANTOR SET No. 2 (FRACTAL MEASURE No. 2)
NUMBER DENSITY
SCALE
CANTOR SET No. 2 (FRACTAL MEASURE No. 3)
NUMBER DENSITY
SCALE
CANTOR SET No. 2 (FRACTAL MEASURE No. 4)
NUMBER DENSITY
SCALE

Figure 34.

Fractal Measure (Cantor Set 1, N=4)

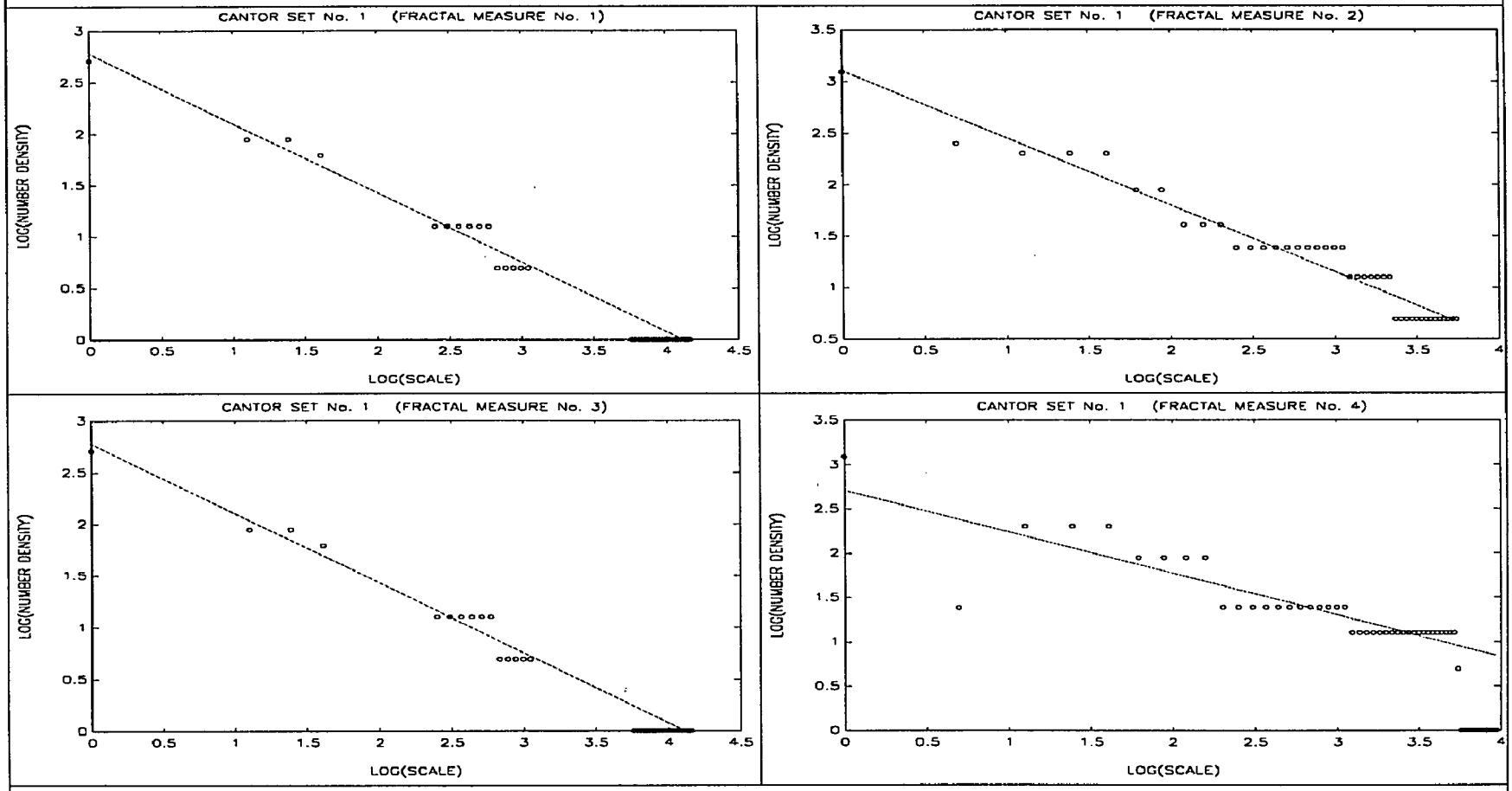

Figure 35.

Fractal Measure (Cantor Set 2, N=4)

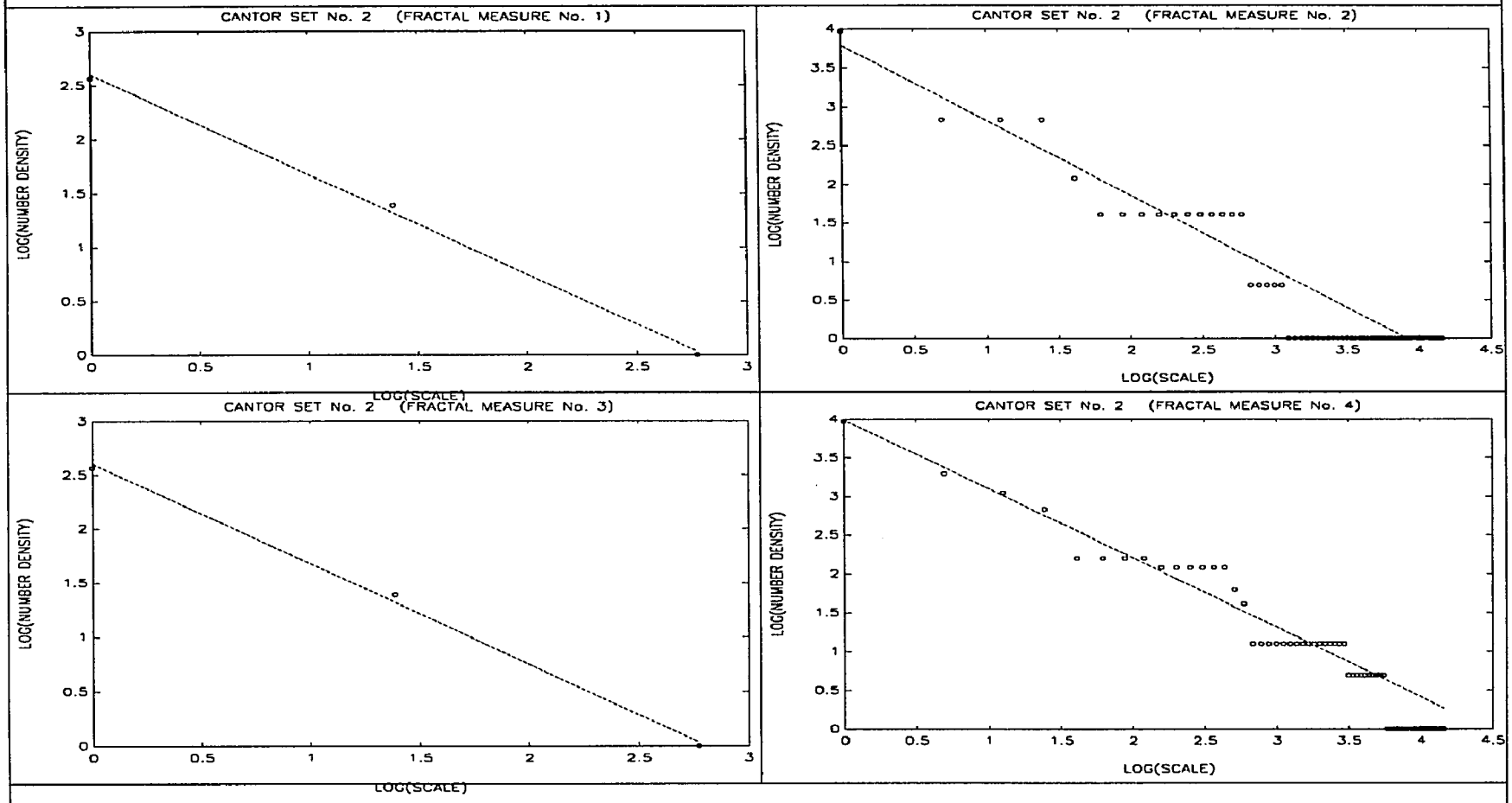

Figure 36.

Polyfractal Measure of Cantor Sets

POLY FRACTAL DIMENSIONS OF CANTOR SETS

	F DIMENSION 1	F DIMENSION 2	F DIMENSION 3	F DIMENSION 4
CANTOR SET No. 1	0.44971	0.67570	0.44971	0.60660
CANTOR SET No. 2	0.40803	0.60608	0.40803	0.81208

Sonlinh Phuvan, 14 January 1993

Table 8.

3.4 Fractal Scanning Technique for 2D to 1D Texture Mapping

3.4.1 Introduction

In the previous sections two new techniques, N-Wavelet coding and Polyfractal measure, for texture identification and classification has been described. The N-Wavelet coding technique can uniquely identify substructures in a texture with added noise. This technique is position and scale invariant, it is also sensitive and noise tolerant. A N increases the noise tolerance increases. Polyfractal measure is used to obtain a vector of fractal measure for each of the subtexture produced by the N-Wavelet Coding. In this section a technique is described which allows the use of 1D N-Wavelet coding and 1D Polyfractal measure to a 2D texture. The main advantage of parallel 1D processing is that it is typically easier to implement electronically and optically than parallel 2D processing.

A 1-D scan which follows Peano's curve to a desired resolution is demonstrated to preserve 2-D proximity relationship and is furthermore shown to be efficient for Wavelet Transform (WT) processing and artificial neural network pattern recognition. This deterministic fractal sampling method can be implemented in real time using optoelectronic scanning. For example, 2-D texture patterns are analyzed by using 1-D Wavelet Transformation (WT).

Those WT coefficients can be fed into a standard backpropagation neural network for pattern recognition. To speed up training time, a top down design which generalize Hopfield's energy landscape approach is given in terms of Mini-Max pattern classifiers.

A space-filling curve with a lower dimensionality that preserves a neighborhood relationship has been an important recurrent theme throughout the last century. It began with Peano's mathematical proof[54] and Hilbert's verification in the last century. Recently, numerous applications have been developed, e.g. the meal on wheel program[6], computer map storage[70] , optical neural network storage[1] of 4-D interconnects within a 3-D photorefractive crystal[66] (with 3/2 fractal dimensionality), and neural networks[73,74,75]. We will demonstrate a fractal scanning technique which can be implemented using optoelectronic methods and is useful for real time neural net pattern recognitions.

A critical element for a solution in pattern detection is how to represent the data in such a way as to facilitate subsequent processing. Some signal processing techniques are more easily processed if the data is in a one dimensional form than in a two dimensional form. Some techniques such as raster scanning and its derivatives do not preserve the adjacency of information (i.e. two points on adjacent rows in a 2D image may be farther apart in the unidirectional or bidirectional raster scan representation then two points which are farther apart in

the 2D image). A complete basis of Peano's curve is introduced for the first time. This basis is used to produce a fractal scan that preserves adjacency information to a greater extent then raster scan techniques.

3.4.2 Fractal Scanning

In order to preserve adjacency in the mapping of a two dimensional tensor image to a one dimensional vector image, the scanning must fill space in a neighborhood before moving to another neighborhood. As can be seen in Fig. 37 the proximity relationship of images is better preserved by a fractal scan then a bidirectional raster scan. The deterministic fractal scan, as scanning density increases, shows self similarity with respect to scales.

To preserve orientation the scan output must produce identical results independent of the scan direction (i.e. top-down, bottom-up, right-left and left-right). This restricts the scan to symmetric orderly scans. The Peano space filling scan is composed of 4 3x3 scan cell primitives (Fig. 38).

In order for the sampling to be self similar for all scales and to preserve orientation at all scales, the scan must be an orderly, symmetric and antichiral fractal scan (Fig. 39).

Given a N x N scan the angular resolution is $N^2/2\text{-}3(N\text{-}1)$ and scale dynamic range is N. The scan sampling is at the correct sampling density when the fractal dimension of the fractal scan is equal or greater than the feature of the image with the largest fractal dimension.

The fractal scanning can be implemented by programming a photocathode tube to scan in the desired sequence (Fig. 40). A computer controls a driver which modulates an electron beam to scan in the desired sequence. The output of the TV tube is then the fractal scanning of the 2D image. By selecting a moderatly fast TV tube this fractal scan can be implemented in real time (video rates).

3.4.3 Wavelet Transform Processing

The Wavelet function must be selected in such a way as to enhance the desired features of the image. In this case the images are binary and thus a bipolar Wavelet function is selected (i.e. the Haar function, see Fig. 41).

The Haar function applied to a bipolar image through a Wavelet transform will extract edge features from a given image (Fig. 42). For each Wavelet transformed image a measure of the number of edges it detects is obtained by

summing all correlation points for a given scale and adjusting the sum with respect to the scale,

$$E(a)=\left(\sum_{b=-\infty}^{\infty}\left|\sum_{n=-\infty}^{\infty}f(x_n)g((x_n-b)/a)\right|\right)\left(\sum_{b=-\infty}^{\infty}\left|\sum_{n=-\infty}^{\infty}g((x_n-b)/a)\right|\right)^{-1} \tag{127}$$

A test example (Fig. 42) is used to test the algorithm. The top figure is a 1-D object comprised of two bars with four edges. The bottom left figure is the representation of a WT using the Haar function with 11 daughters (WT over 12 different scales). The bottom right figure is the graph of the measure of the number of edges versus the scale (i.e. sampling density or resolution).

It can be seen from Fig. 42 that as the scale increases the measure of the number of edges initially stays constant then monotonically decreases. This is because the sampling resolution becomes inadequate to measure the desired features. This implies that the sampling resolution can be adjusted in such a way as to discrimate against undesirable features.

This monotonic decrease of the measure of edges with increasing scale, and the asymptotic approach to a limit as the scale becomes finer is similar to the way in which fractal dimensions are measured. The fractal dimension typically found in the limit of the ratio of the log of the number of features measured with the log

of the scale used to measure those features as the scale becomes very small. This implies that the fractal dimension can also be measured as follows,

$$D_F = \lim_{a\to 0}\left(\log \sum_{b=-\infty}^{\infty}\left|\sum_{n=-\infty}^{\infty} f(x)g((x_n-b)/a)\right|\right)\left(\log \sum_{b=-\infty}^{\infty}\left|\sum_{n=-\infty}^{\infty} g((x_n-b)/a)\right|\right)^{-1} \quad (128)$$

thus the fractal dimension of an image is contained within its Wavelet transform.

A fractal scan for six different simple geometric patterns is performed and the output is processed via a Wavelet transform. The six geometric figures can be subdivided into two classes; periodic array of diagonal lines and stacks of circles. Within each class the thickness and number of features are varied.

In Figs. 45-48, the upper left graph is the scan pattern and the 2D input image, the lower left graph is the 1D result of the fractal scan, the upper right graph is the WT of the 1D output of the fractal scan using the Haar function over 12 scales (i.e. one mother and 11 daughters), and the lower right graph is the plot of the measure of the edges with respect to the scale used during the WT processing.

It can be seen that as the thickness of the diagonal lines increases, certain periodicity in the Wavelet transform becomes apparent (Figs. 43-48). The

measure of the number of edges decreases monotonically as the scale increases (Figs. 43-48).

It can be seen from Fig. 43-48 that the Wavelet transform using the Haar wavelet function produces distinctly different output for both the intraclass and interclass images. This implies that the Wavelet transform of a modified Peano fractal scan using the Haar wavelet function can be used for highly discriminatory 2D pattern classification. A candidate for classification is to feed those wavelet coefficients into a multiple layer artificial neural network with a standard backward error backpropagation training algorithm. However, in order to speed up training time, we considered in Sect. 2.3.4 other approaches.

3.4.4 Artificial Neural Net Classification

3.4.4.1 Designs of Energy Cost Functions in a Neuronic Vectorial Representation

An important question for practical applications is how to speed up the training process and to ensure a fast convergence of weight adjustment. We have suggested a general procedure of Taylor series expansion of the clustering-declustering mini-max energy to estimate the synaptic weights. Here, we extend the procedure by a self-consistently variational technique to make the truncated higher order terms of the Taylor series negligible.

A top-down design of a hard-wired neural network algorithm has been initiated by Hopfield et al for constrained optimizations. We consider a supervised top-down design goes beyond Hopfield's attempt. The minimum clustering of the alike and the maximum de-clustering of the dis-alike seems to be two contradicting goals. A tradeoff can be mathematically constructed by the linear combination of those pairs alike in the numerator and the pairs of dis-alikes in the denominator of the mini-max energy formalism (schematically shown in the cost energy expression as follows)

3.4.4.2 Design of Hard Wired Neural Network Mini-Max Energy Principle

To speed up training time, a top down design which generalize Hopfield's energy landscape approach is given in terms of mini-max pattern classifiers. The mini-max energy function[73] is defined to be

$$E(f_c) = a \sum_{c \neq c'} \langle f_c | f_{c'} \rangle + b \sum_{c=1,2} |f_c - I_c|^2 + \sum_{c \neq c'} \frac{d}{|f_c - f_{c'}|} \qquad (129)$$

where I is a template of the c-class = 1,2 together with feature f, and the coefficient of the direction cosine via inner product < > may be heavily weighted, eg. by setting a =10 (relative to b = 1, and d = 10). Because the template image is fractal scanned to preserve the neighborhood proximity relationship, a Taylor

series expansion along the Peano curve can be used to derive multiple layer interconnects. While the first order derivative is reserved for the neurodynamics equations, the second order derivative is evaluated at the equilibrium state: $V_i = V_i^{(0)}$, $V_j = V_j^{(0)}$ becomes the Taylor's coefficient, identified as a hard-wired Hopfield interconnect weights,

$$T_{i,j}^{(0)} = \left(\frac{\partial^2 E}{\partial V_i \partial V_j} \right)_{V_i^{(0)} V_j^{(0)}} \tag{130}$$

Then, the Hopfield-like hard-wired interconnect $T_{i,j}^{(0)}$ becomes softwired by means of the Hebbian learning that make the cubic order negligible.

$$T_{i,j} = T_{i,j}^{(0)} + \varepsilon \Delta V_i \Delta V_j \quad \text{where} \quad T_{i,j,k} \Big|_{V_i = V_i^{(0)} + \Delta V_i, V_j = V_j^{(0)} + \Delta V_j} << T_{i,j}^{(0)} \tag{131}$$

Similarly, the procedure can be extended to three layers.

3.4.5 Results

A fractal scan has been introduced which preserves orientations and proximity relationships. Once texture images are fractal scanned, the scanned binary template are fed to determine feature vectors by using Cauchy simulated annealing technique[73,74,75]. Fig. 49 shows two extracted features from two texture inputs, thin diagonal lines and thicker circles. The diagonal feature output does

not have pixels at the center, which make it orthogonal to the other feature. In general, when an object is sampled above the Nyquist rate (the size of the Peano scan cell is compatible with the size of the object), the feature output would be more distinguishable. The hetero-associative memory, constructed between input and its feature, is used to classify these two classes as follow.

$$f_{c'} = \sum_{c=1}^{2} [f_c I_c^T] I' \tag{132}$$

The Wavelet transform of the fractal scan produces distinct output for both intraclass and interclass images implying possible uses as input for subsequent classification processes such as artificial neural networks.

A relationship between the edge measures obtained for the Wavelet transform using the Haar function and the fractal dimension has been implied.

A real time optoelectronic implementation of the fractal using computer controlled TV tubes has been proposed.

3.4.6 Conclusions

A fractal scanning method has developed which preserves 2D proximity relationships and orientation. The complete basis for making Peano curves is introduced.

This technique is very well matched for use in Wavelet transform processing because of its self similarity with respect to scale and because its 1D vector data representation simplifies Wavelet tranform analysis. The Wavelet transform of a 1D vector produces a 2D output, but that of a 2D tensor produces a 4D output, this makes it difficult to analyze. The Wavelet transform is essentially a correlation of a reference function with an arbitrary waveform in which the size of the reference function is changed with respect to scale (i.e. a correlation at various scales). The self similarity with respect to scale of the fractal scan match the variation of scale of the Wavelet transformation.

Using the nonlinear classification and feature extraction properties of artificial neural networks, operating on the output of the Wavelet transform analysis of the fractal scanned image, may produce a high discrimination and high sensitivity pattern detection and classification technique.

One potential classification technique is to use the Mini-Max type artificial neural networks to find orthogonal Wavelet basis functions to classify image textures. The Mini-Max algorithm is ideally suited for this purpose since it tries

to maximize interclass differences and minimize intraclass differences (i.e. orthogonalize classification features). This may subsequently allow us to produce an alphabet of texture primitives for texture classifcation.

Vector Scans

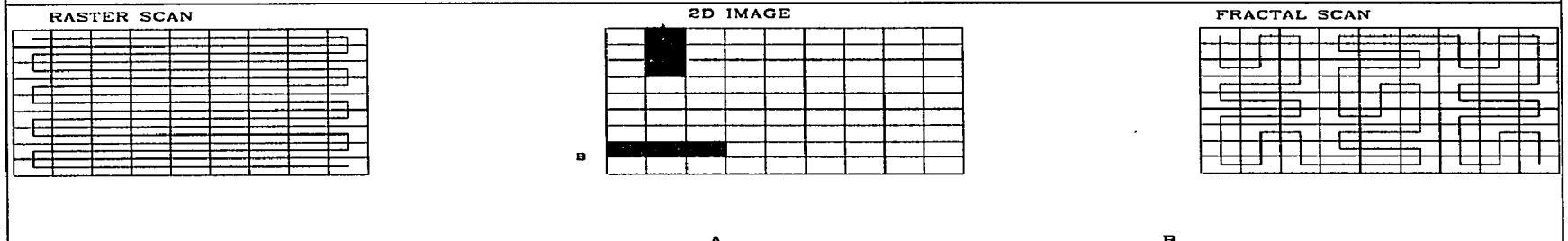

Figure 37.

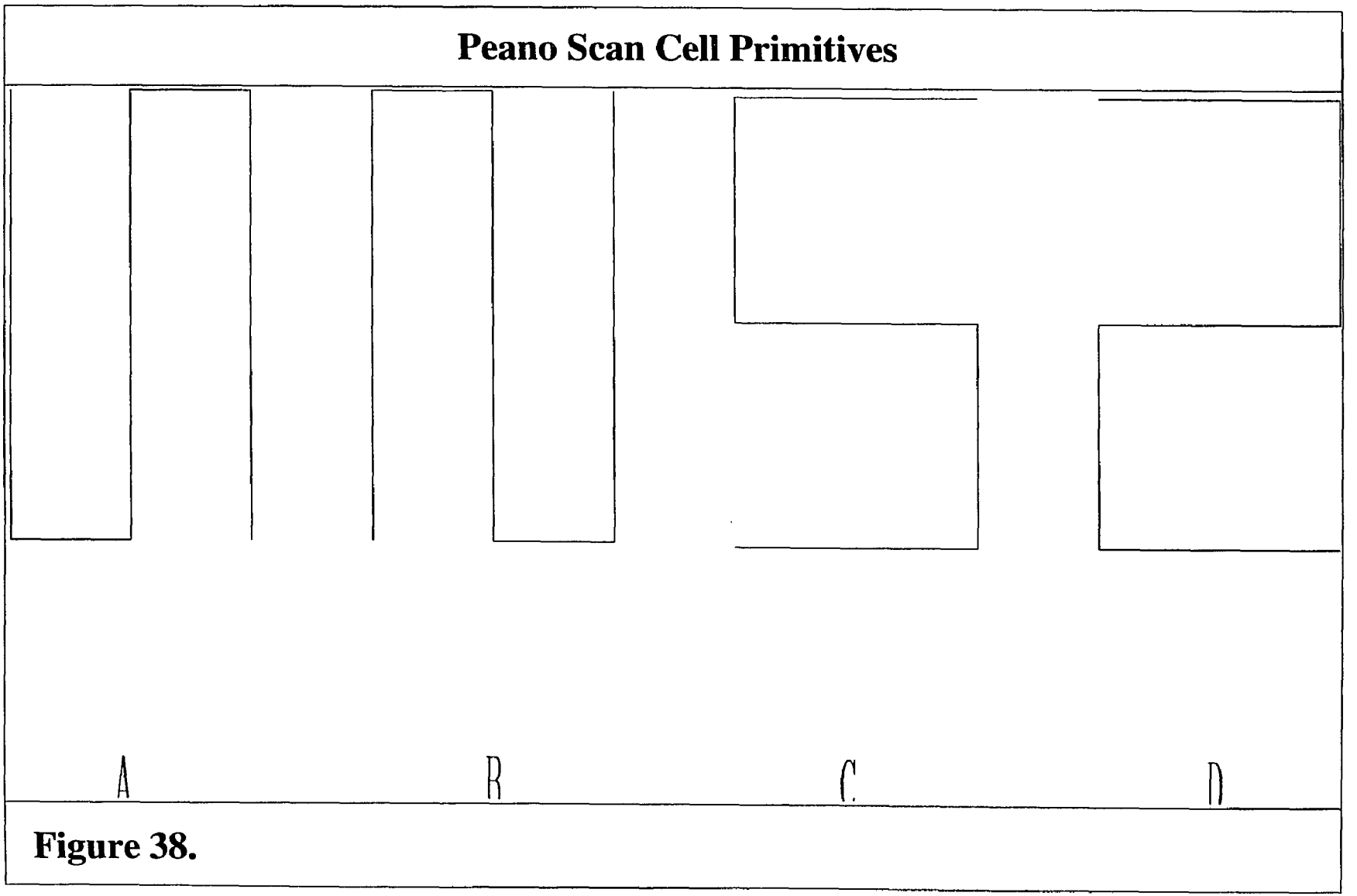

Figure 38.

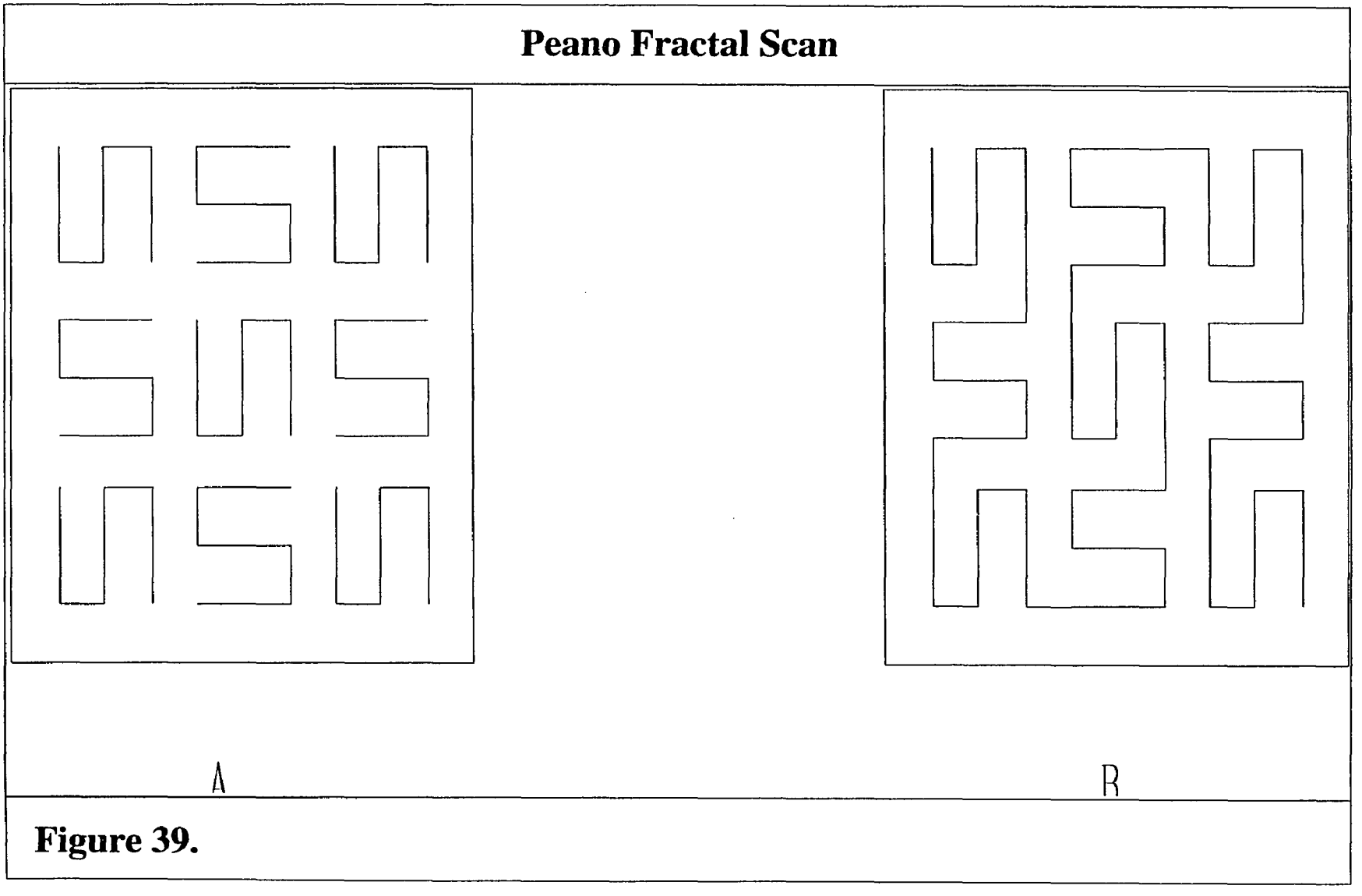

Figure 39.

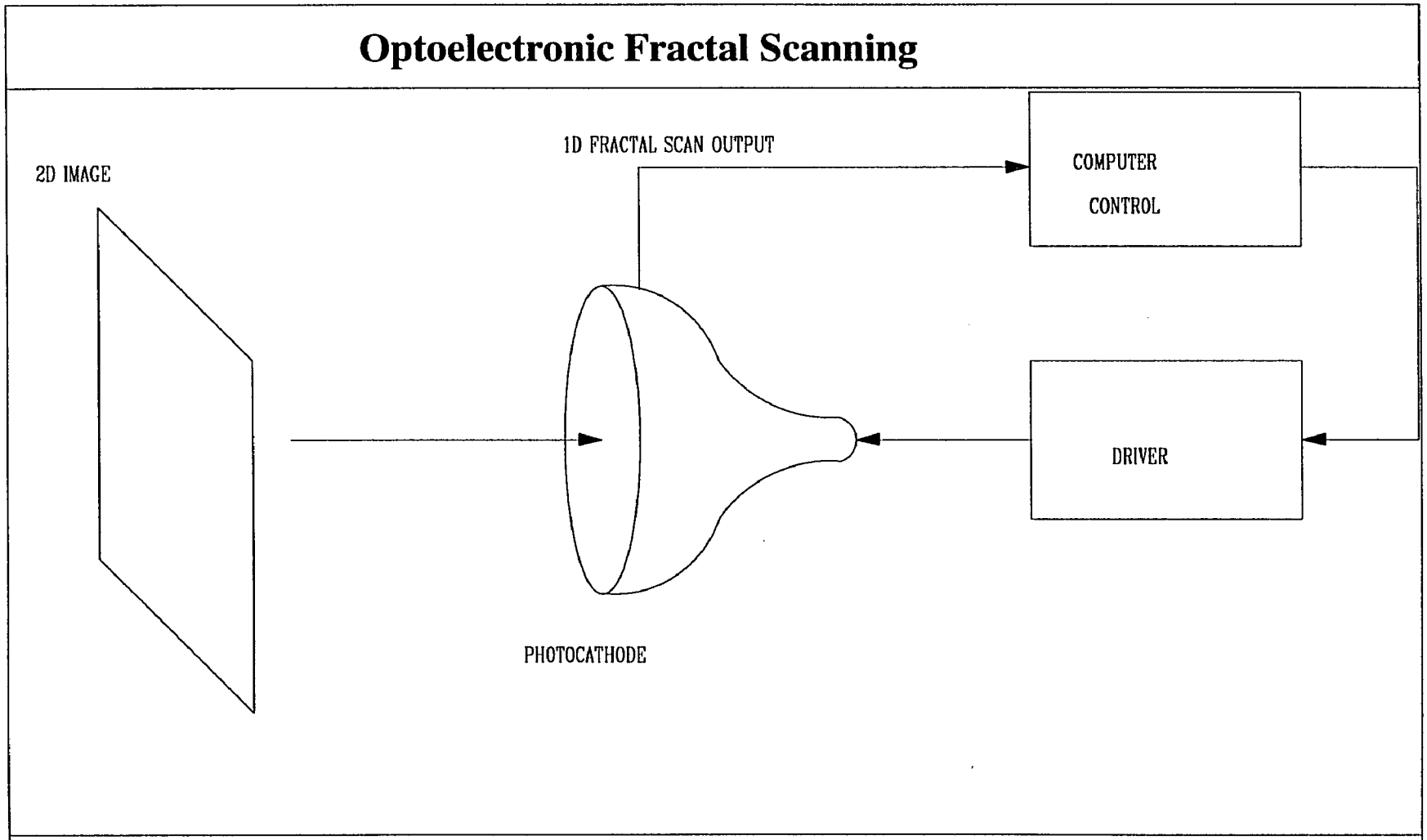
Optoelectronic Fractal Scanning
2D IMAGE
1D FRACTAL SCAN OUTPUT
COMPUTER
CONTROL
DRIVER
PHOTOCATHODE

Figure 40.

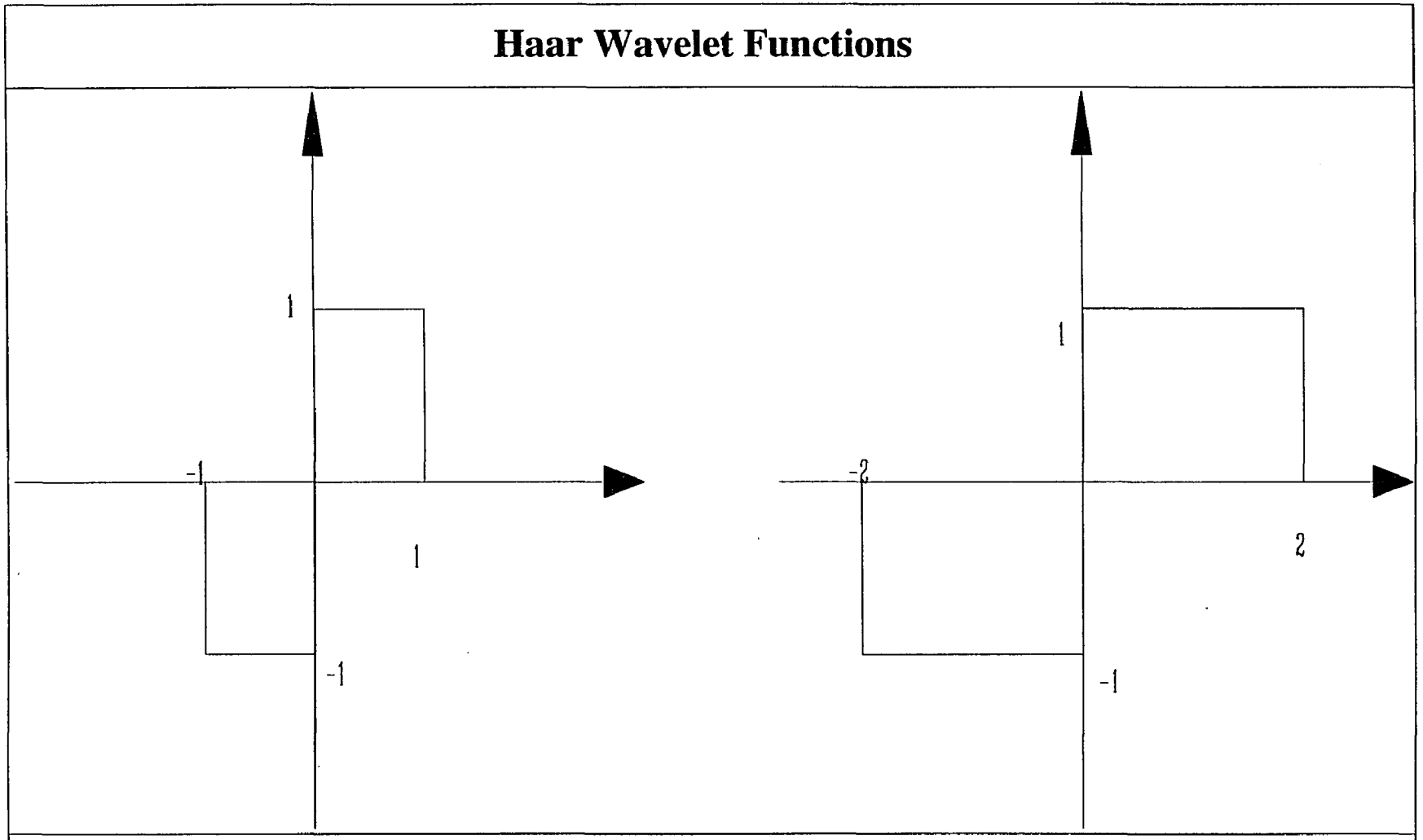

Figure 41.

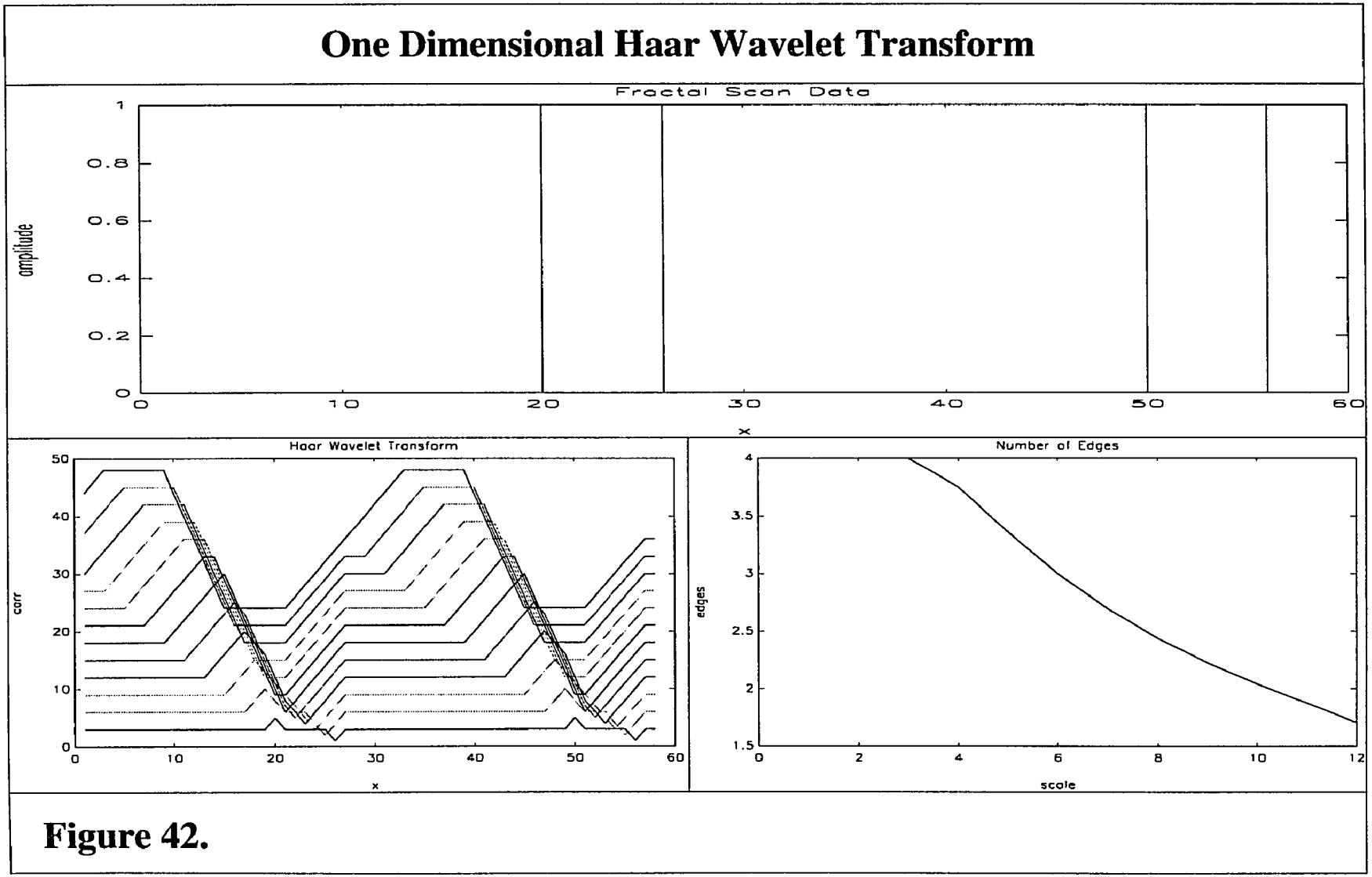
One Dimensional Haar Wavelet Transform
Fractal Scan Data
amplitude
x
Haar Wavelet Transform
corr
x
Number of Edges
edges
scale

Figure 42.

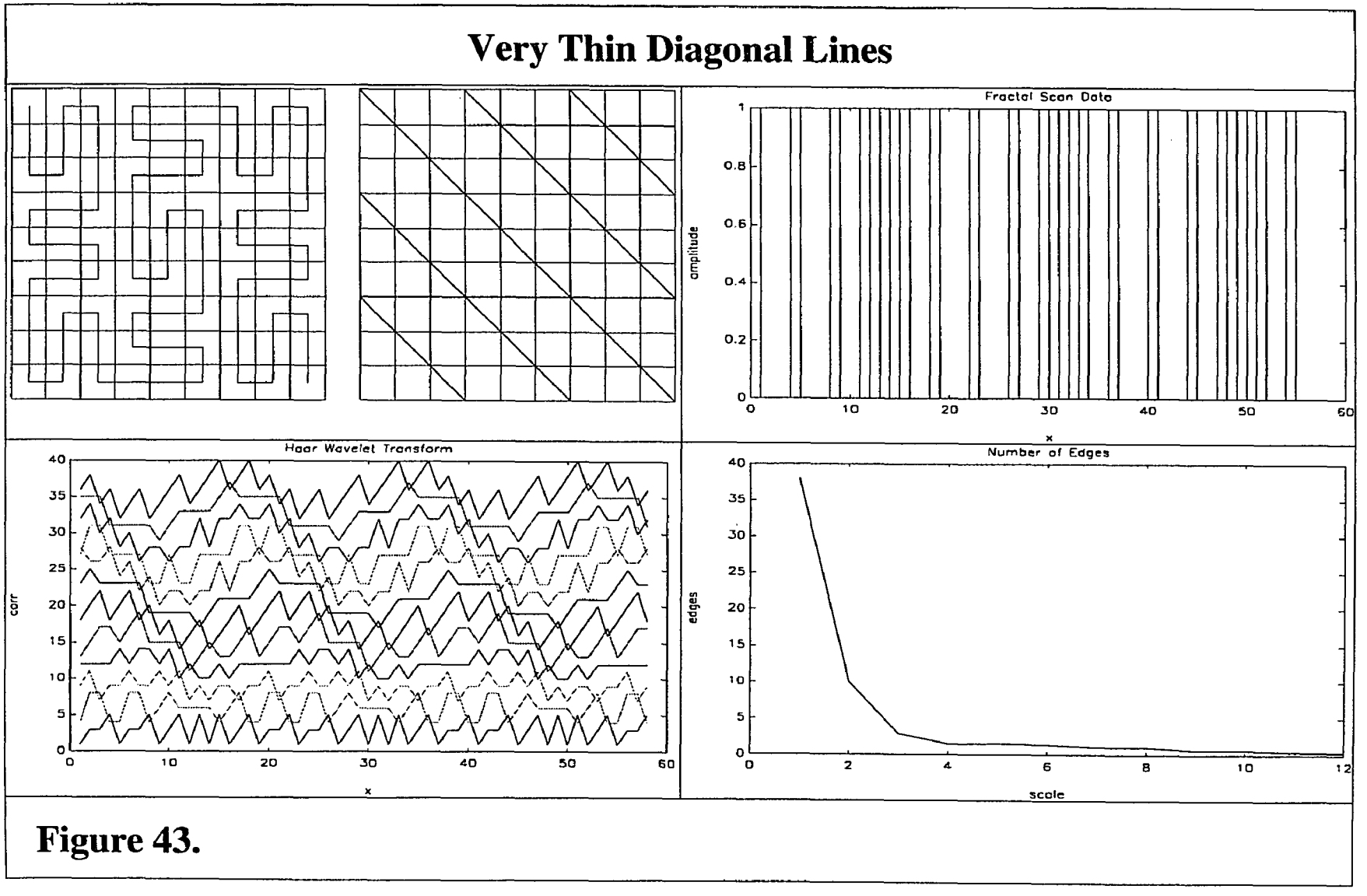
Very Thin Diagonal Lines
Fractal Scan Data
amplitude
x
Haar Wavelet Transform
corr
x
Number of Edges
edges
scale

Figure 43.

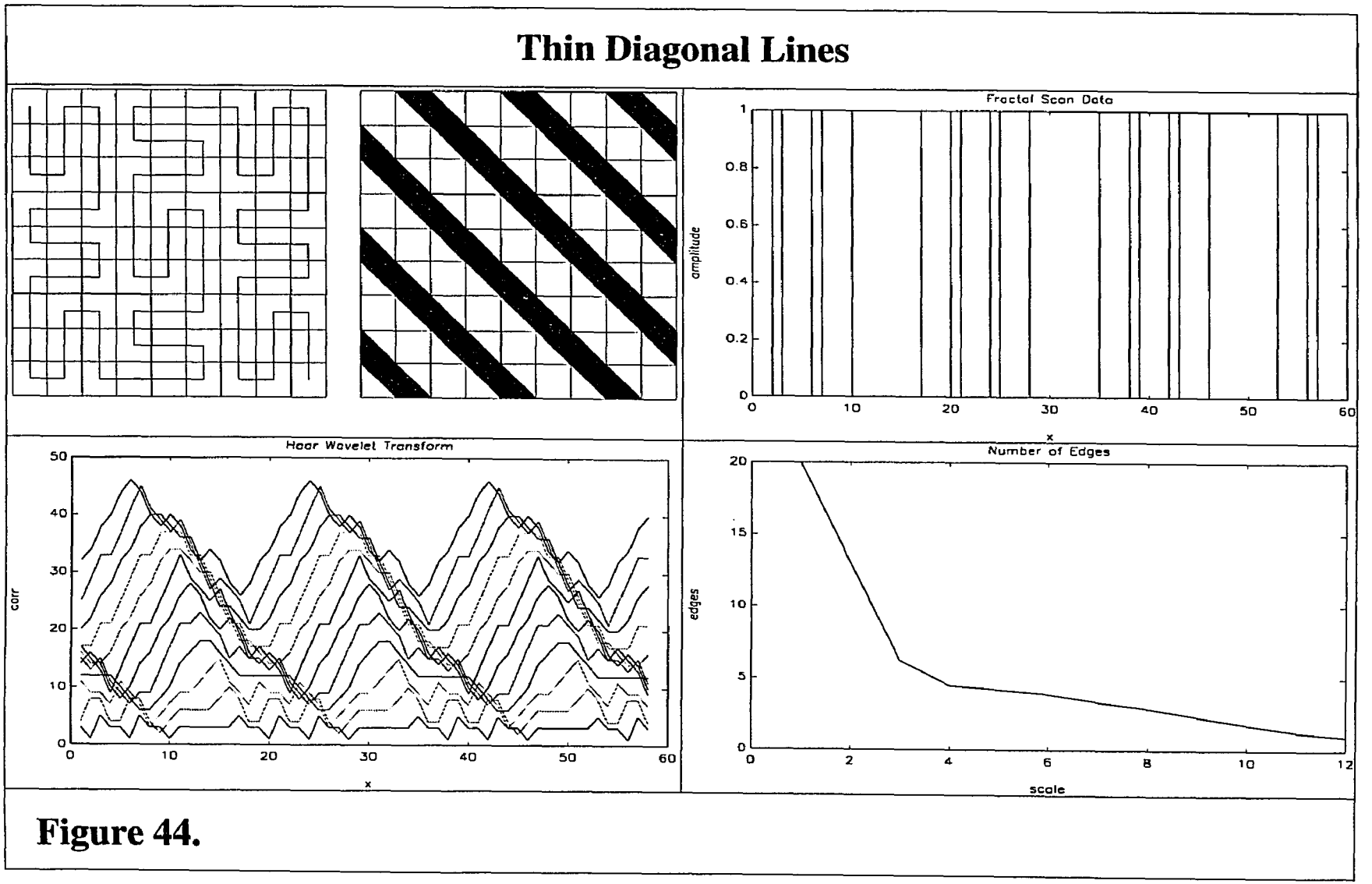
Thin Diagonal Lines
Fractal Scan Data
amplitude
x
Haar Wavelet Transform
corr
x
Number of Edges
edges
scale

Figure 44.

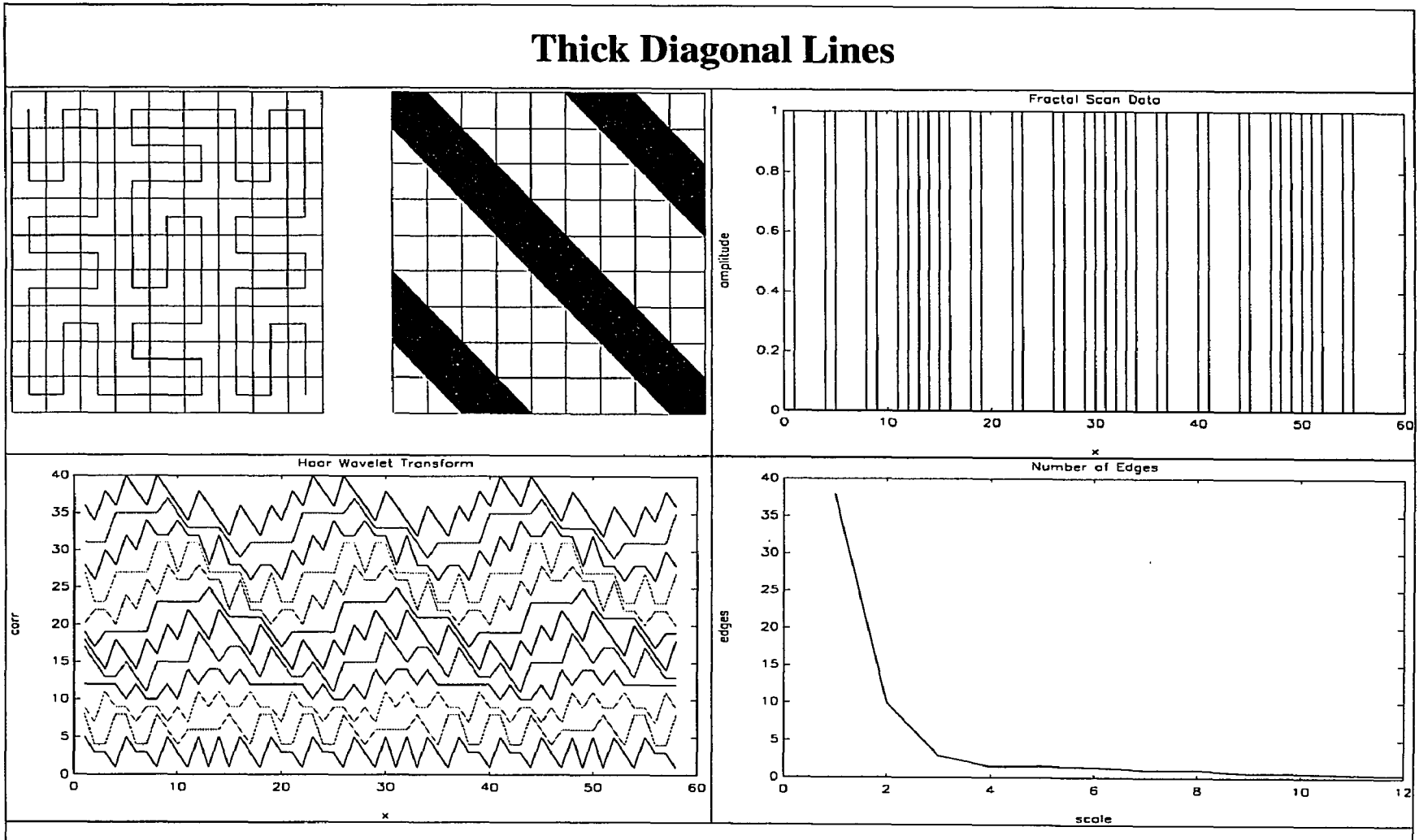
Thick Diagonal Lines
Fractal Scan Data
amplitude
x
Haar Wavelet Transform
corr
x
Number of Edges
edges
scale

Figure 45.

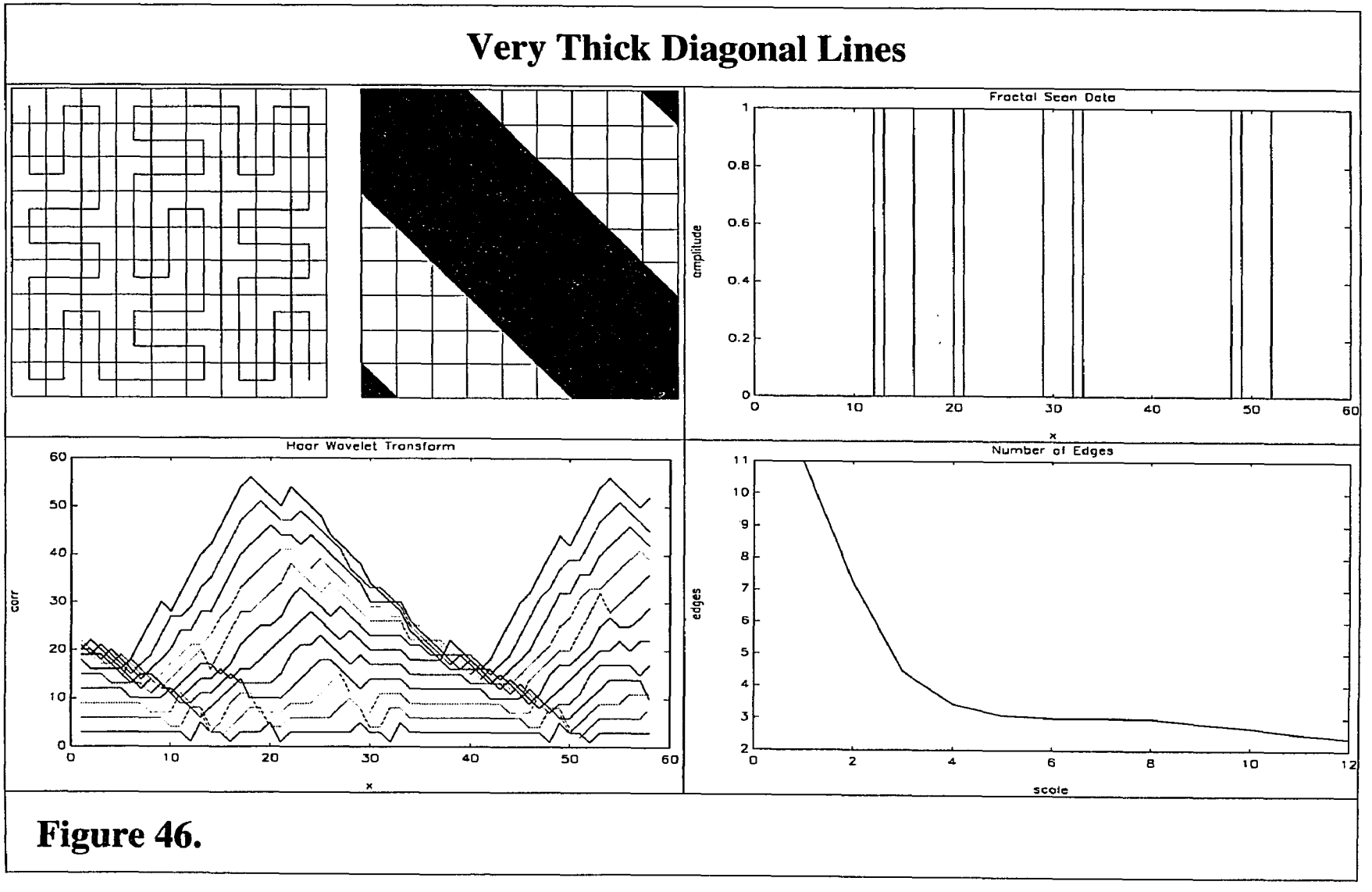
Very Thick Diagonal Lines
Fractal Scan Data
amplitude
x
Haar Wavelet Transform
corr
x
Number of Edges
edges
scale

Figure 46.

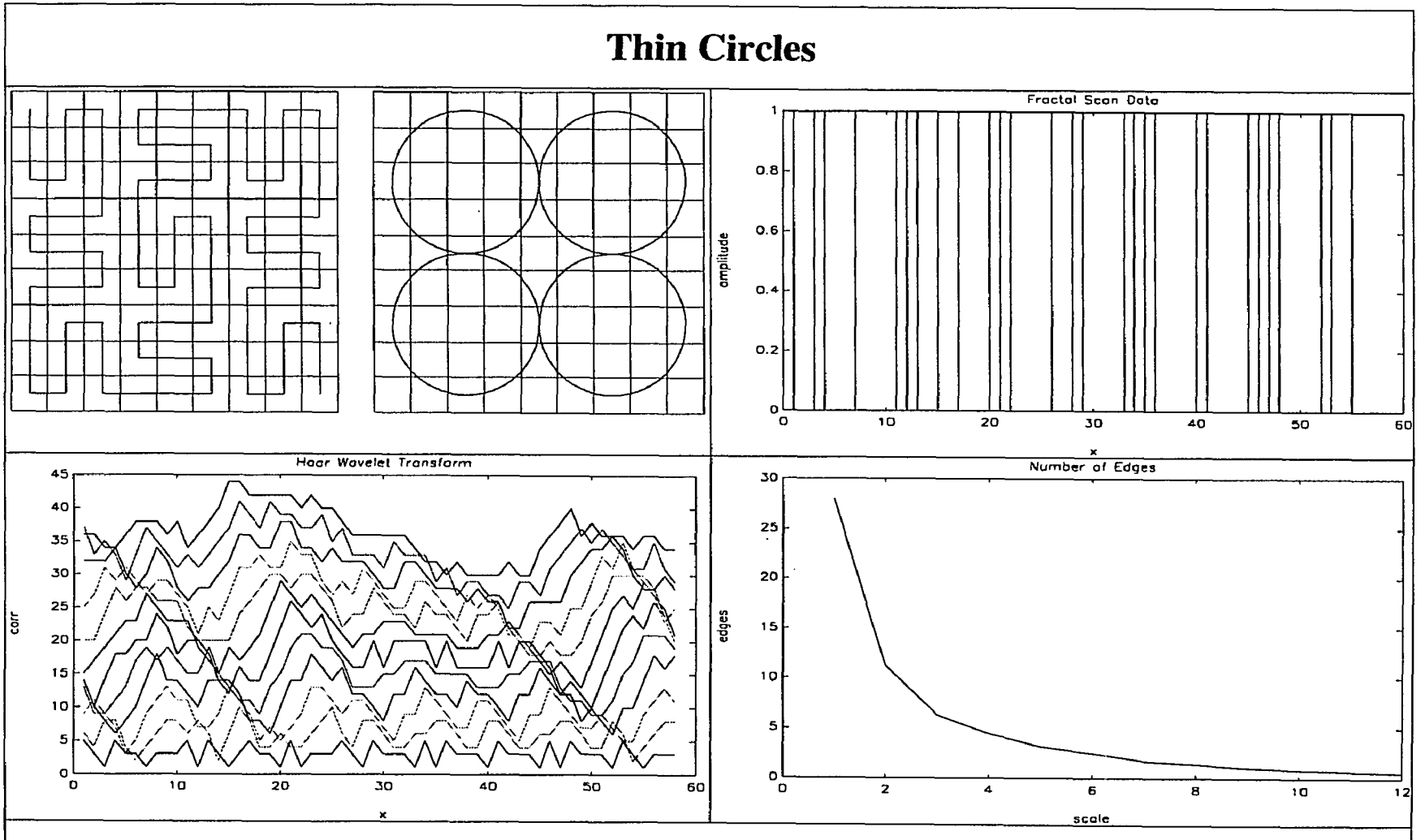
Thin Circles
Fractal Scan Data
amplitude
x
Haar Wavelet Transform
corr
x
Number of Edges
edges
scale

Figure 47.

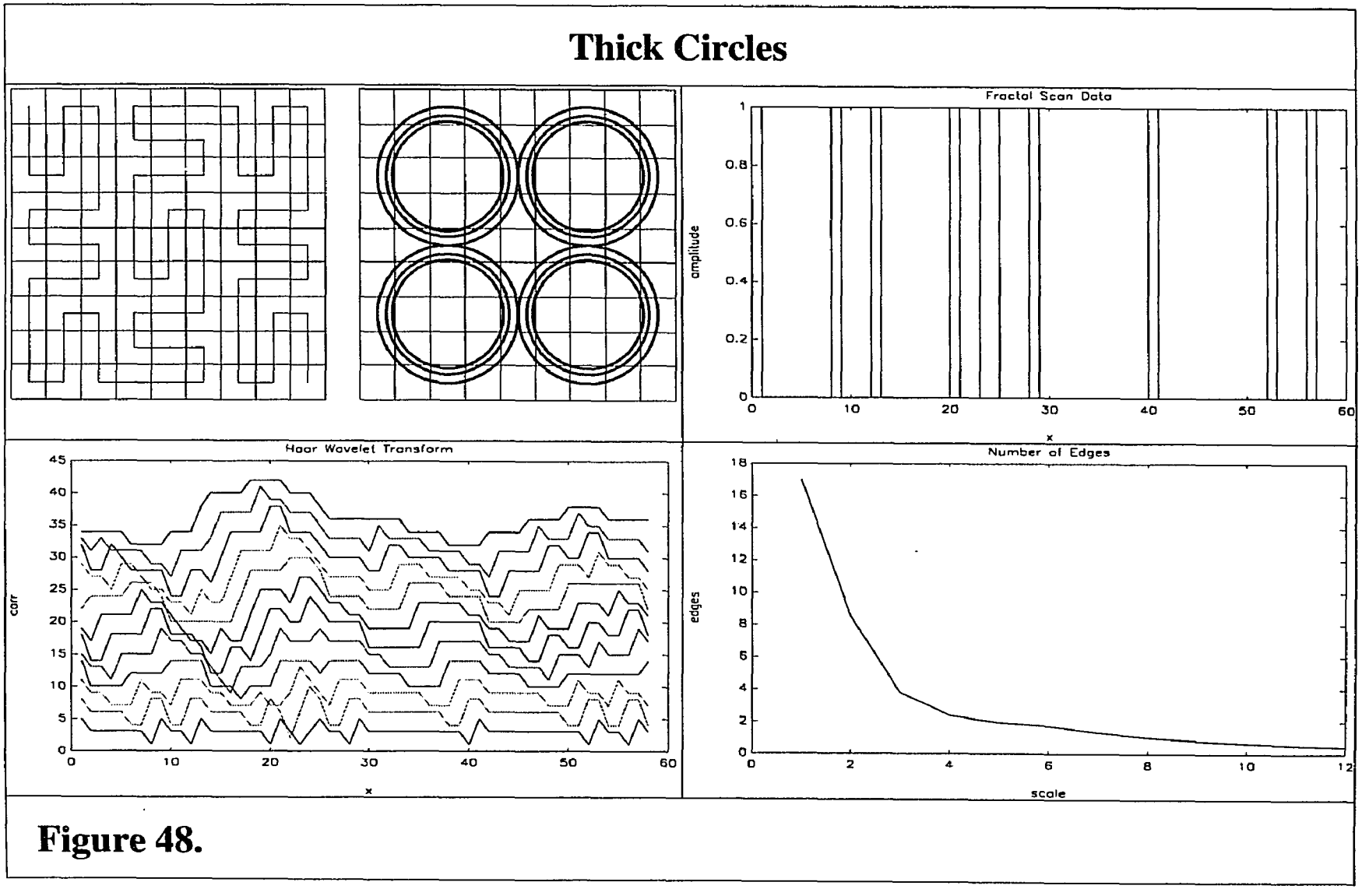
Thick Circles
Fractal Scan Data
amplitude
x
Haar Wavelet Transform
corr
x
Number of Edges
edges
scale

Figure 48.

3.5 Conclusions

In section 3.2 a new detection and classification technique, N-Wavelet coding, has been described. The detection and classification is position and scale invariant. By increasing the number of wavelets, N, used in the classification process, it is shown that it is possible to decrease the probability of false alarm rate for a given probability of detection and noise, and distortion level.

In section 3.3 a new technique for characterizing texture, Polyfractals, is described. This technique provide a vector of fractal measure, each fractal measure quantifies the relationship between the incidence of a specific substructure within a texture and the sampling scale used to detect the discriminating features of a substructure. This technique can provide a unique polyfractal even for textures with identical fractal dimension.

In section 3.4 a new technique, Fractal Scanning, is used to produce a scale invariant mapping of 2D texture into a 1D texture. This allows 1D processing for the classification of 2D texture, thus simplifying the design and implementation of real time classification using the N-Wavelet and polyfractal techniques.

4 Optical Implementation

4.1 Overview

In the previous chapter, a novel method for characterizing texture has been demonstrated. In this chapter a method for implementing the algorithms developed in that chapter is described.

There are three major operations which need be executed in order to implement the polyfractal algorithm. A fractal scan, a wavelet transform and a fractal measure. There are several techniques for implementing the fractal scan. A electronically controlled camera can be used, or a normal raster scan camera can be remapped digitally to a fractal scan. The 1D optical wavelet transform has been previously implemented[88] using acousto-optic modulators and the 2D optical wavelet transform has been implemented using a film and a mechanical stage[20]. The fractal measure has also been implemented optically[20], but not in real time.

In this chapter a different approach is used. In both the wavelet transform and the fractal measure, the operations can be either be performed in parallel or/and serially. An architecture based on fast binary optically addressed, electrically controlled ferroelectric spatial light modulators is described. This architecture is capable of concurrent parallel and serial operations.

As defined in the previous chapter, the wavelet transform can be thought as a correlation with respect to changes in scale of the wavelet function used to perform the correlation. In this architecture it is proposed that the wavelet function is inserted as a modulated optical pulse into an optical ring with unity gain. With every ring rotation the optical pulse is scaled through a set of lenses, and the resultant wavelet function is correlated with an input function to produce the wavelet transform. Several wavelet functions can be spatially multiplexed onto a single optical pulse, thus allowing concurrent parallel and serial operations.

Since it is desired to implement the polyfractal algorithm in real time, binary optically addressed electrically controlled ferroelectric spatial light modulators are used. Those spatial light modulators are fast, the frame rate can exceed 50kHz. A major drawback of this type of light modulators are that they are binary, i.e. they are completely on or off. Optically addressed binary spatial light modulators are usually faster then their grey level counterparts by several order of magnitudes. Implementation of the correlation operation in the wavelet transform requires grey level processing. It is shown that real time grey level processing can be obtained by using a halftoning technique.

The optical iterative processor described can thus be used to implement an optical wavelet transform and obtain a polyfractal measure. An optical implementation for estimating the fractal dimension of two dimensional images has

been previously implemented. That implemention is based on an iterative operation where the scaling of the sampling function is implemented by moving an optical filter incorporating a wavelet function along an optical track and lens set.

In both the execution of the wavelet transform and the fractal measure, and iterative process is required. The optical iterative processor described in the following sections can be used to implement those operations in real time.

4.2 Grey Level Optical Processing from Binary OASLM using Halftoning

4.2.1 Introduction

Wavelet transform processing is grey level, but many optically addressed spatial light modulator are binary. Optically addressed binary spatial light modulators are usually faster then their grey level counterparts. The frame rate difference is typically several order of magnitudes higher for the binary SLMs.

Theoretically, a ferroelectric OASLM can have several hundred pixel per milimeter resolution, but so far only about 120 pixels/mm has been demonstrated. Potentially, a ferrolectric OASLM frame rate can approach a 100 kHz frame rate, but a 15 kHz has been demonstrated. Still binary OASLMs far outperform grey level OASLMs.

In this section it will shown that it is possible to get grey level processing from a binary spatial light modulators using halftoning techniques, thus optical wavelet processing can then be implemented using binary level spatial light modulators. The use of halftoning techniques reduces the effective resolution of OASLMs. The reduction in resolution is linearly proportional to the dynamic range of the grey level desired.

A hydrogenated amorphous silicon ferroelectric liquid crystal (ASFLC) spatial modulator has been used as a thresholding device to produce a real time optical logarithmic transform at frame rates up to 12 kHz. This is a significant improvement over previous work in which a frame rate of 50 Hz was obtained using a Hughes light valve. The logarithmic transfer function reported in this paper occurs for input intensity between 4 to 9 mW. It is shown that significant increase in the input dynamic range can be obtained by modification of the halftone screen and of the ASFLC SLM used as a thresholding device. A comparison of theoretical modeling with experimental data has shown close agreement.

Nonlinear transformations are used in many signal processing applications, such as automated target recognition, pattern matching, guidance, control, and artificial neural networks. Presently, the majority of those nonlinear

transformations are implemented electronically. Electronics are inherently serial and thus impose constraints on the speed at which data can be processed. An alternate way to process the data is in parallel. Optics because of its inherent parallellism is a good candidate technology for attaining that goal. Optical implementation of linear transformations is much simpler than that of nonlinear transformations.

One nonlinear transformation, the logarithm, is used in cepstrum[1,2,48,52] and MLF (Mean Log Fourier) processing. The cepstrum and the MLF are nonlinear signal processing techniques used in image and radar processing.

The optical log has been previously implemented in non-real time using photographic film and liquid crystal light valves[2,3-8,12,13,14,34,45]. An optical log, demonstrated at 10 kHz using an Amorphous Hydrogenated Silicon Ferroelectric Liquid Crystal (ASFLC) Spatial Light Modulator (SLM), is described in the subsequent sections.

4.2.2 Optical Logarithm

The log transform can be implemented in real time by using a halftone screen in conjunction with a spatial light modulator (SLM) with tresholding capability.

A halftone screen is a transparency composed of a regular array of halftone cells. The halftone screen characteristics are determined by the optical density of each halftone tone cell and the cell density. The periodicity of the halftone cells, L, is chosen such that the highest spatial frequency component in the input image has a much larger periodicity than that of the halftone cells.

The output intensity I_0 can be described by the local input picture intensity, I_{in}, if the input intensity varies slowly with respect to the spatial sampling rate of the halftone screen.

The amplitude transmittance, t_h, can be expanded into a complex Fourier series:

$$t_h(x,y) = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} B_{k,l} e^{-j2\pi kx/L} e^{-j2\pi ly/L} \tag{133}$$

where

$$B_{k,l} = \int_0^L \int_0^L B_{k,l} e^{-j2\pi kx/L} e^{-j2\pi ly/L} dxdy \tag{134}$$

Each of the coefficients of the Fourier transform of the transmittance function appears as isolated spectral islands. Those spectral islands correspond to the spatial harmonics of the halftone cells in the halftone screen. Using a spatial

filter, a single island or spectral order can be selected. The resulting intensity distribution at the output is then given as

$$I_0(I_{in},k,l) = |B_{k,l}|^2 = \left| \frac{1}{L^2} \int_0^L \int_0^L t_h(x,y) e^{-j2\pi kx/L} e^{-2\pi ly/L} dxdy \right|^2 \tag{135}$$

The transmittance function can be expressed as a function of exposure E and the optical density profile of the halftone cell $f(x,y)$,

$$t_h(x,y) = g(\log E) \tag{136}$$

where

$$E = I_{in} 10^{-f(x,y)} \tag{137}$$

When only the zeroth order of the Fourier transformed halftone image is used, then $k,l = 0$, and equation becomes

$$I_0(I_{in},0,0) = \left| \frac{1}{L^2} \int_0^L \int_0^L g(\log I_{in} - f(x,y)) dxdy \right|^2 \tag{138}$$

If we assume that the halftone cell is circularly symmetric and that the transmittance is monotonically decreasing from the origin ($x,y = 0$) but keep a

rectangular area of integration (Figure 49) to juxtapose the halftone cells to produce the regular array of cells comprising the halftone screen, we obtain

$$I_0 = \frac{1}{L^2}\left|\int_0^L \int_0^L g\left(\log I_{in} - f\left(\sqrt{x^2+y^2}\right)\right)dxdy\right|^2 \tag{139}$$

since

$$\begin{aligned} x &= r\cos(\theta) \\ y &= r\sin(\theta) \\ z &= \sqrt{x^2+y^2} \end{aligned} \tag{140}$$

we obtain

$$I_0 = \frac{1}{L^2}\left|\int_0^L \int_0^L g(\log I_{in} - f(r)rdrd\theta)\right|^2 \tag{141}$$

If we assume that the transmittance function is that of an ideal binary recording medium, i.e., with perfect thresholding characteristics and threshold intensity I_r,

$$g(\log E) = \left\{ \begin{array}{ll} a & \log I_{in} - f(r_0) < \log I_r \\ b & \log I_{in} - f(r_0) \geq \log I_r \end{array} \right\} \tag{142}$$

There are two cases, $r_0 \leq L$ and $r_0 > L$.

For $r_0 \leq L$ we obtain,

$$I_0 = \frac{1}{L^2}\left|\int_{\pi/4}^{\pi/2}\left[\int_{r_0}^{L/\sin(\theta)} ardr + \int_0^{r_0} brdr\right]d\theta + \int_0^{\pi/4}\left[\int_{r_0}^{L/\sin(\theta)} ardr + \int_0^{r_0} brdr\right]d\theta\right|^2 \quad (143)$$

using the bifold symmetry about $\theta = \pi/4$,

$$I_0 = \frac{1}{L^2}\left|2\int_0^{\pi/4}\left[\int_{r_0}^{L/\cos(\theta)} ardr + \int_0^{r_0} brdr\right]d\theta\right|^2 \quad (144)$$

scaling r_0 with L to produce a dimensionless position $z = r_0/L$, we obtain,

$$I_0 = \left|2\int_0^{\pi/4}\left[\int_{z_0}^{1/\cos(\theta)} azdz + \int_0^{z_\theta} bzdz\right]d\theta\right|^2 \quad (145)$$

$$I_0 = \left|\int_0^{\pi/4}\left[a\left[\frac{1}{\cos^2(\theta)} - z_0^2\right] + bz_0^2\right]d\theta\right|^2 \quad (146)$$

$$I_0 = \left|a + \frac{\pi}{4}z_0^2(b-a)\right|^2 \quad (147)$$

To obtain the halftone cell optical density profile for a logarithmic transformation let

$$I_0 = K\log\left(\frac{I_{in}}{I_r}\right) \quad (148)$$

and since

$$f(z_0) = \log (I_{in}/I_r) \tag{149}$$

then the halftone cell optical density profile is

$$f(z) = \frac{1}{K}\left[a + \frac{\pi z^2}{4}(b-a)^2\right]^2 \tag{150}$$

where $z \leq 1$.

For $r_0 > L$ we obtain,

$$I_0 = \left|2\int_0^{\theta_0}\int_0^{1/\cos(\theta)} bzdzd\theta + \int_{\theta_0}^{\pi/4}\int_0^{z_0} bzdzd\theta + \int_{\theta_0}^{\pi/4}\int_{z_0}^{1/\cos(\theta)} azdzd\theta\right|^2 \tag{151}$$

where $\theta_0 = \cos^{-1}(1/z_0)$

$$I_0 = \left|a + (b-a)\tan(\theta_0) + (b-a)\left(\frac{\pi}{4} - \theta_0\right)z_0^2\right|^2 \tag{152}$$

from which we obtain the optical density profile of the halftone cell,

$$f(z)=\frac{1}{K}\left[a+(b-a)\tan\left(\cos^{-1}\left(\frac{1}{z}\right)\right)+(b-a)\left(\frac{\pi}{4}-\cos^{-1}\left(\frac{1}{z}\right)\right)z^2\right]^2 \qquad (153)$$

where $1 \leq z \leq 1/cos(\theta)$.

The density profile for an ideal binary thresholding function along the bifold axis of symmetry at $\pi/4$ is shown in Figure 50.

Thus, by placing the logarithmic halftone screen behind the input image, subsequently exposing the SLM with the resulting pulse modulated input image, thresholding the SLM, passing the zeroth order diffraction and reading the SLM with a readout beam, an optical logarithmic transform of the initial image is obtained.

4.2.3 Optical Architecture

The experimental setup is as shown in Figure 51. The output beam of an air cooled argon laser is expanded and collimated. The collimated beam then is modulated by a Kodak contact screen. The contact screen is composed of a two dimensional matrix of halftone cells with a cell density of 100 dots per inch. The pulse modulated output of the contact screen is imaged onto the write side of the ASFLC SLM for subsequent thresholding. The thresholded output is read by a HeNe laser at the output face of the SLM. The zeroth order diffraction pattern is

obtained by low pass filtering the thresholded pulse modulated image through a Fourier lens set. The zeroth order output is then the logarithm of the input image.

4.2.4 Components Characterization

4.2.4.1 Kodak Halftone Screen

An image processing workstation is used to digitize a halftone supercell in the Kodak contact screen. A supercell is comprised of four halftone cells juxtaposed in such a way as to produce a continuous supercell. A transmissivity plot of a halftone supercell is obtained in two orthogonal directions (Figure 52). It is assumed that the supercell is circularly symmetric. As can be seen from Figure 52, the density cell is nonuniform due to the graininess of the film used. The ratio of the maximum to minimum intensity is approximately 3.8 .

4.2.4.2 ASFLC SLM

The ASFLC SLM used is a prototype fabricated by Displaytech under a Navy contract. The cross section of the optically addressed ASFLC SLM is shown in Figure 53. The principal elements are a p-i-n diode juxtaposed on a layer of ferroelectric liquid crystal (FLC) and sandwiched between two layers of

transparent conducting oxide (TCO). The TCO is used to produce a voltage potential across the p-i-n diode and the FLC layer. The p-i-n diode is light sensitive and light impinging upon it will produce a charge distribution at the interface between the p-i-n diode and ferroelectric liquid crystal. This voltage potential distribution in turn produces a director distribution in the FLC layer which in turn produces a phase distribution in the light reflected from the ferroelectric liquid crystal layer by the p-i-n diode/FLC interface. The current prototype lacks a dielectric mirror.

The active area of the ASFLC SLM is approximately 1 cm in diameter. This area is pixelated into a 6 by 6 matrix (Figure 54). The purpose of this pixelation is to enable bypass of any shorts produced by inhomogeneities in the amorphous silicon layer. The device operates by subjecting the p-i-n diode/FLC sandwich to an alternating square wave potential. A negative potential enables writing of the device and a positive potential erases the device. It is found that different pixels have differing optimal frequencies and bias. This is due to the inhomogeneities produced during the manufacture of the device. Partly due to the lack of a dielectric mirror at the diode/FLC layer it is found that a maximum contrast ratio of 4:1 can be obtained. The resolution is approximately 100 lines per mm, which varies due to inhomogeneities.

The thresholding characteristic is shown in Figure 55. It was found that better thresholding was obtained at higher operating frequency, near 10 kHz, than a lower frequency such as 2 kHz. As the operating frequency was increased the peak to peak voltage was also increased to 45 Volts to optimize the thresholding characteristics of the device. Only one pixel was used to obtain the threshold curve to minimize the effect of inhomogeneities. The equivalent film gamma is 5.2.

The utilization of FLC in the device allows higher resolution and frame speed then SLM using nematic and smectic liquid crystals as well as some devices using electro-optic and magneto-optic materials (Figure 56).

4.2.5 Modeling and Simulation

4.2.5.1 Halftone Cell

The experimentally obtained halftone cell profile is replotted as a function of the log of the transmittance, $\log(t_h(z))$, versus the dimensionless position, z. The two profiles obtained from the two orthogonal cuts are then split into four segments. These four profile segments are subsequently used to fit a tetradic least square curve (Figure 57),

$$f(z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4 \pm 0.12 \tag{154}$$

where,

$$\begin{aligned} a_0 &= 0.19 \\ a_1 &= 0.67 \\ a_2 &= 1.10 \\ a_3 &= -2.11 \\ a_4 &= 0.88 \end{aligned} \tag{155}$$

As shown in Figure 57, the fit follows the general curve of the experimentally obtained cell profile data points with an uncertainty of ± 0.12. The roughness of the transmissivity profile of the halftone cell is due to its manufacture. The halftone cells were obtained from a commercially available contact screen with a supercell density of about 100 dots/inch. At the halftone cell level, the grain boundaries of the film is clearly visible and produce the inhomogeneities shown in Figure 52. These inhomogeneities may introduce a bias in the fit of the halftone cell density profile, which in turn may produce a bias in the predicted output transfer function.

4.2.5.2 Thresholding Characteristic

The thresholding characteristic of the ASFLC (Figure 55) is used to fit a sigmoid curve of the form,

$$g(z) = \frac{b_0}{1 + e^{-b_1(z-b_2)}} + b_3 \tag{156}$$

where,

$$\begin{aligned} b_0 &= 0.35 \\ b_1 &= 9.27 \\ b_2 &= -0.03 \\ b_3 &= 0.46 \end{aligned} \tag{157}$$

The sigmoid model fits closely the experimentally determined threshold data (Figure 58). The maximum transmissivity of the ASFLC is 0.81 and the minimum transmissivity is 0.46. This provide a contrast ratio of 1.76, which is very low. The low transmissivity is due in part to the inhomogeneities in the device as well as the high frame used to obtain a sharp threshold.

4.2.5.3 Simulation

Substituting the fitted halftone cell density profile, $f(z)$, and threshold characteristic of the ASFLC, $g(z)$, in the intensity function,

$$I_0 = \left| 2 \int_0^{\pi/4} \int_0^{1/\cos(\theta)} g(\log I_{in} - f(z) z dz d\theta) \right|^2 \quad (158)$$

and numerically integrating it over the area of the cell, we obtain the predicted transfer function (Figure 59). The values of the output intensity vary from 0.21 to 0.65. The output intensity remains constant below 0.6 and above 0.8 corresponding to the sensitivity and saturation limits of the ASFLC SLM. The usable input range to obtain a log transform is between 0.6 and 0.8.

4.2.5.4 Experimental Results

The optical log is obtained by varying the input laser intensity and plotting against the output intensity on a semilog graph (Figure 60). A linear fit using regression analysis is used to determine whether a log has been obtained and over what range.

Between 0 to 4 mW no response is obtained due to the intrinsic thresholding characteristic of the ASFLC SLM. From 5 to 9 mW the curve is linear and a logarithmic transfer function is obtained within experimental error. Beyond 10 mW the device saturates.

The input beam intensity must be between 5 to 9 mW to obtain a logarithmic transfer function.

A reduction in the inhomogeneity of the ASFLC SLM will increase the sharpness of the thresholding function and decrease experimental variations.

The theoretical model and the experimental data are in close agreement except for a *20%* shift in the domain of the input intensity (figure 61) which is within the uncertainty in the measurement of the halftone density profile (figure 57). Two contributing factors exists, the scattering in the measurement of the halftone cell optical density which can produce a constant bias in the model, and the limited sampling density of the thresholding device which has the effect of approximating the continuous halftone cell transmittance by a discrete staircase profile.

To reduce the experimental variation in the log transfer function the halftone screen must be smoothed out. At this time the halftone cells are grainy and so consequently is the density profile.

A reduction in the inhomogeneity of the ASFLC SLM will increase the sharpness of the thresholding function and decrease experimental variations.

The spatial sampling rate of the ASFLC SLM can be increased by increasing the size of the halftone cells or by increasing the resolution of the ASFLC SLM.

The dynamic range of the present system is small, less than an order of magnitude. However, with improvements in the ASFLC SLM and the halftone tone screen it should be possible to increase the dynamic range by at least one order of magnitude. Because the SLM is optically addressed and the input beam and the output beam are physically distinct, two dynamic ranges can be considered; the input dynamic range and the output dynamic range.

The input dynamic range (D_{in}) can be defined as the ratio of the maximum to minimum input beam intensity required to reach the saturation point of the SLM.

$$D_{in} = \frac{I_{in,\max}}{I_{in,\min}} \tag{159}$$

since

$$I_{in,\max} t_{h(\min)} = I_{in,\min} t_{h(\max)} = I_{in,saturation} \tag{160}$$

then,

$$D_{in} = \frac{t_{h(\max)}}{t_{h(\min)}} \tag{161}$$

and since,

$$t_{h(\min)} = 10^{-f_{\max}}$$
$$t_{h(\max)} = 10^{-f_{\min}} \tag{162}$$

we obtain

$$D_{in} = 10^{f_{\max} - f_{\min}} \tag{163}$$

thus an increase in the optical density range of the halftone screen of one will increase the input dynamic range by a factor of ten.

The output dynamic range (D_{out}) can be defined by the ratio of the maximum average output intensity to the minimum average output intensity.

$$D_{out} = \frac{I_{out,\max,avg}}{I_{out,\min,avg}} \tag{164}$$

since

$$I_{out,avg} = \frac{1}{A_{cell}} \int_0^{A_{cell}} I_{out,iniyial} R(A) dA \tag{165}$$

where, $R(A)$ is the contrast ratio of the SLM output

$$D_{out} = \frac{A_{cell} R}{A_{on,\min} R + A_{off}} \tag{166}$$

The output dynamic range then increases as the resolution of the SLM increase since this would decrease $A_{on,min}$. The output dynamic range also depends on R to some extent since an increase in the contrast ratio will decrease the relative contribution of A_{off}. Increasing A_{cell} will increase D_{out} but will decrease the image sampling density. Therefore it is better to increase resolution since this will increase the output dynamic range without loss in the image sampling density.

A further consequence of increasing the resolution of the SLM is to allow a more accurate sampling by the SLM of each cell profile in the halftone screen. This will provide for a smoother transfer function between input and output intensity.

A halftone screen with a profile matched to the sigmoid threshold characteristic of the ASFLC SLM would produce a better logarithmic fit.

4.2.6 Conclusions

A real time optically implemented logarithm was obtained albeit with low dynamic range and a *20%* domain shift between experimental and predicted results (Figure 136).

The domain shift with respect to predicted result may be resolved by using a more uniform halftone screen (Equation 154).

The input dynamic range can be increased by using a halftone screen with a larger optical density range then presently being used (Equations 159-163).

The output dynamic range can be increased without sampling density penalty by increasing the resolution of the optically addressed SLM (Equation 164- 166).

An improved SLM is presently being developed by Displaytech under a Navy contract (Figure 56).

Ongoing efforts are presently being made to produce a halftone screen with a higher optical density range and a more ideal cell density profile.

Figure 49.

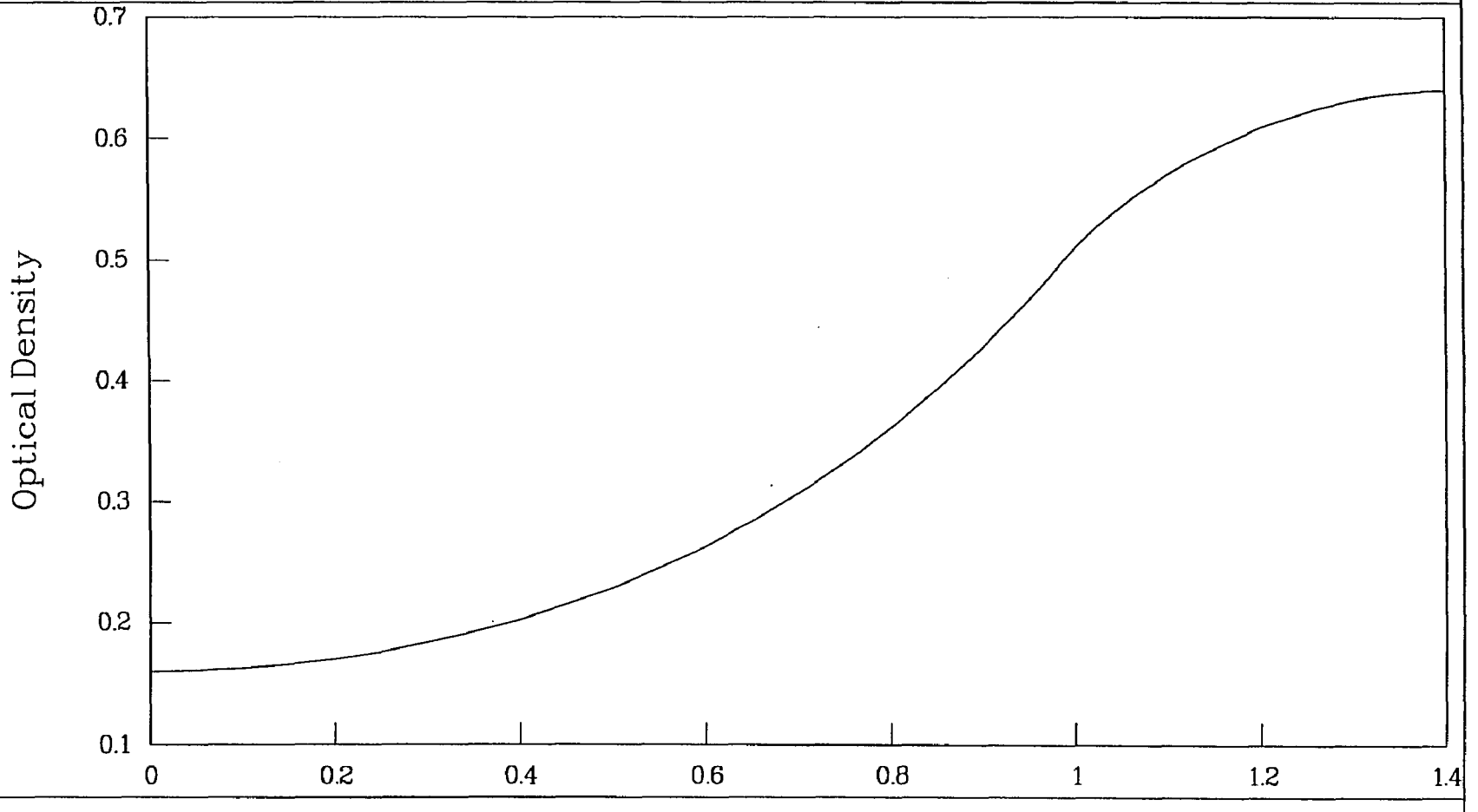
Ideal Halftone Density Profile for Ideal Binary Thresholding
Optical Density
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.2
0.4
0.6
0.8
1
1.2
1.4

Figure 50.

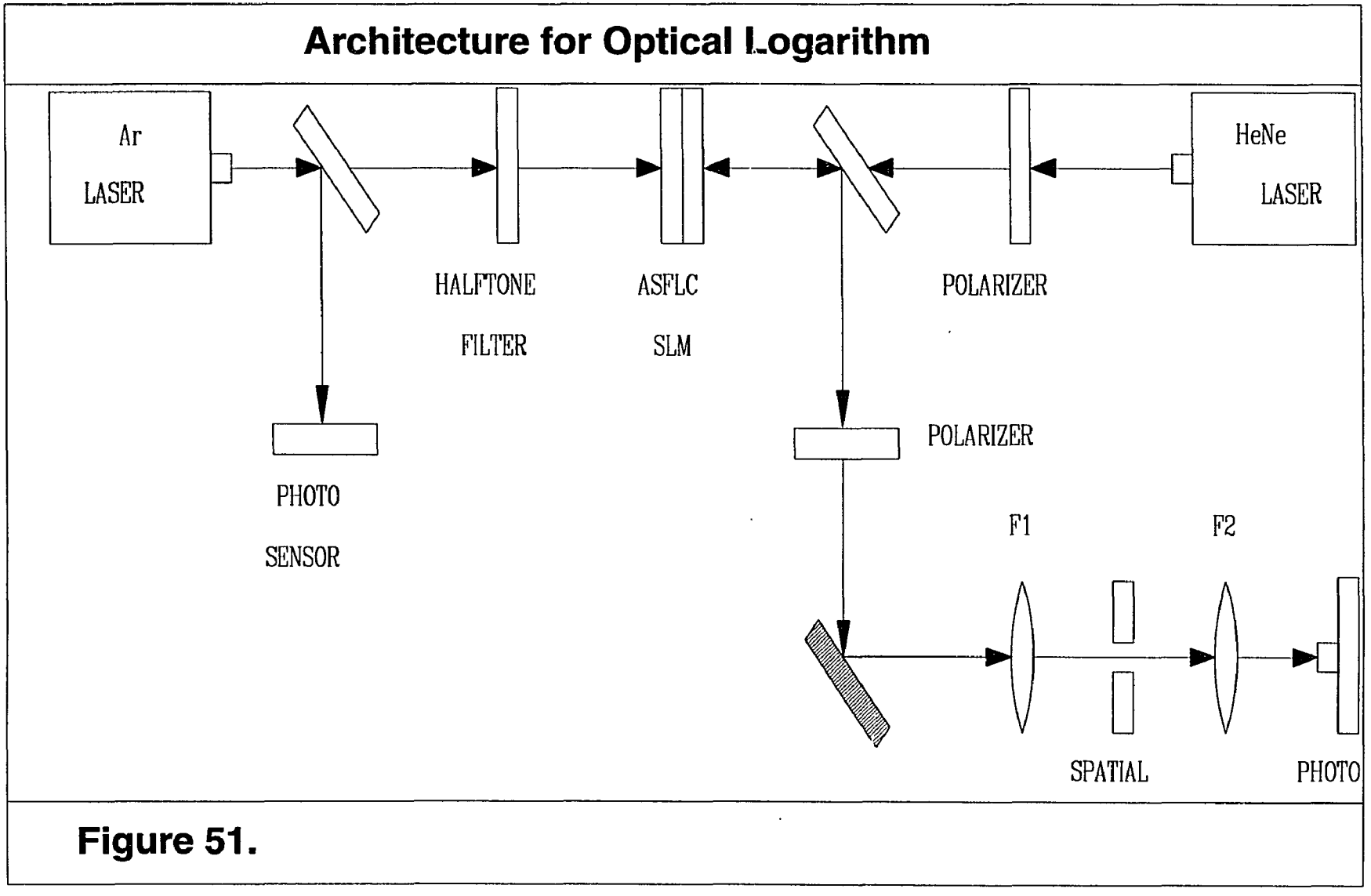
Architecture for Optical Logarithm
Ar
LASER
HeNe
LASER
HALFTONE
FILTER
ASFLC
SLM
POLARIZER
PHOTO
SENSOR
POLARIZER
F1
F2
SPATIAL
PHOTO

Figure 51.

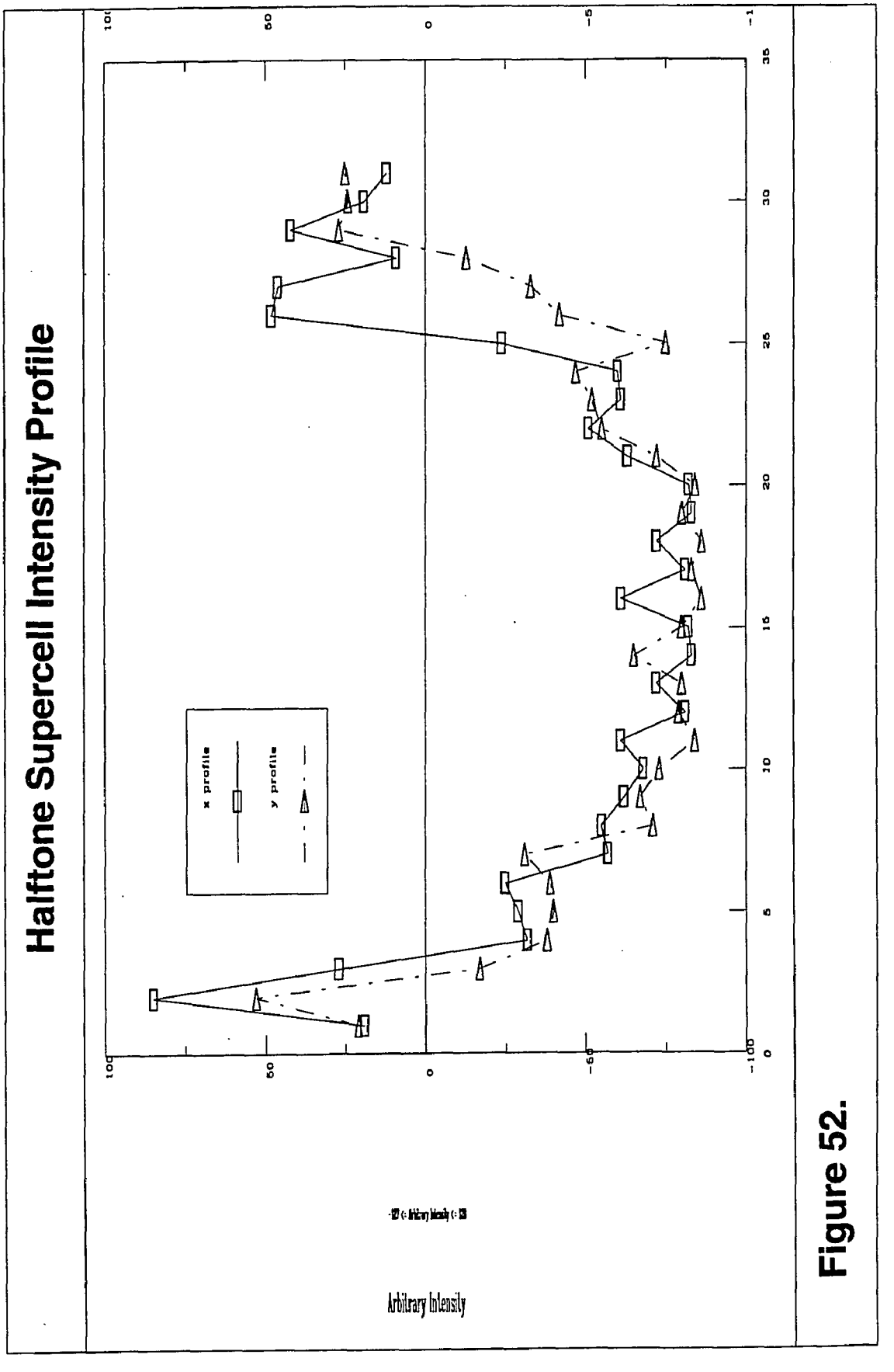
Halftone Supercell Intensity Profile
x profile
y profile
Arbitrary Intensity

Figure 52.

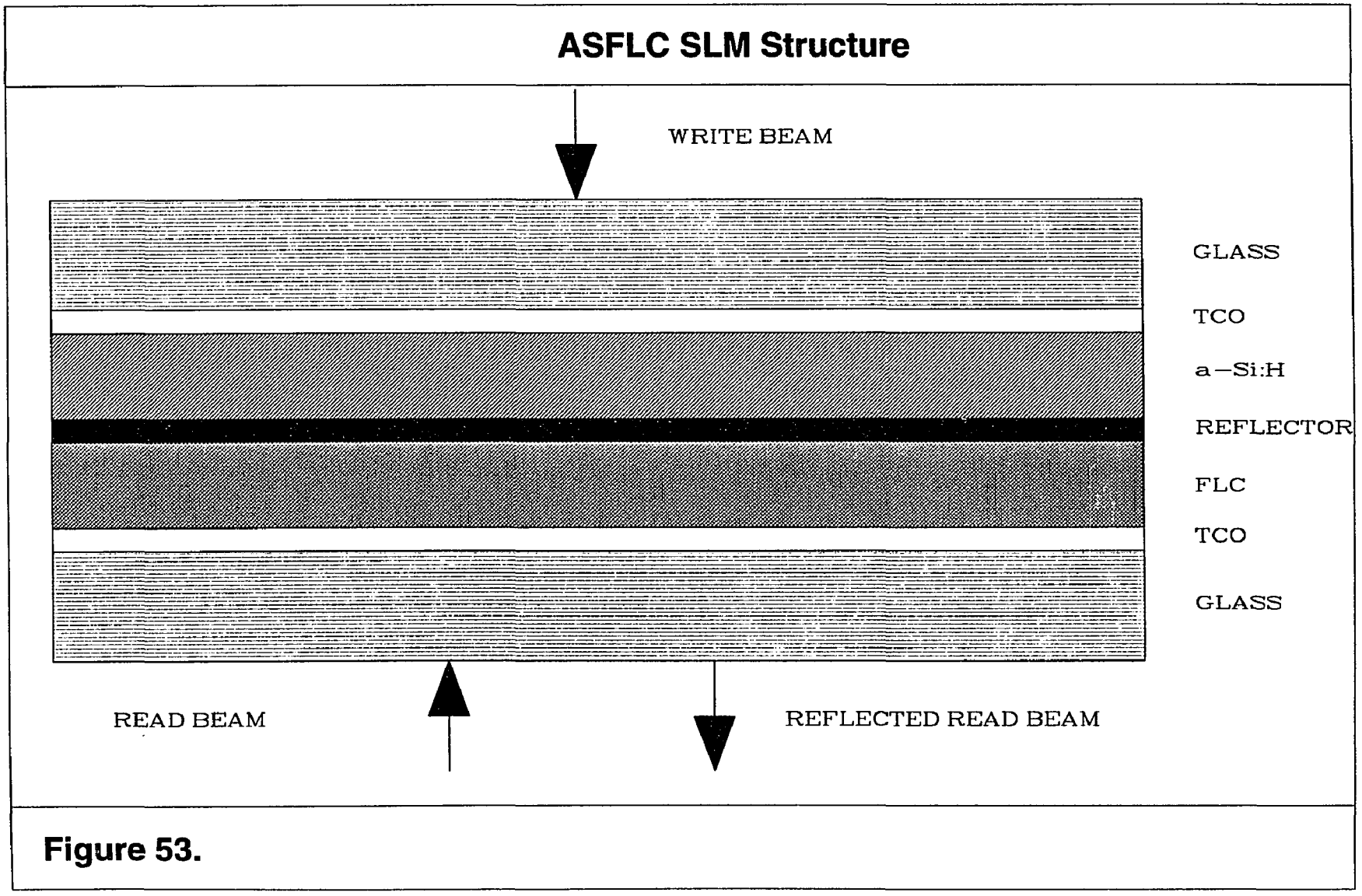
ASFLC SLM Structure
WRITE BEAM
GLASS
TCO
a–Si:H
REFLECTOR
FLC
TCO
GLASS
READ BEAM
REFLECTED READ BEAM

Figure 53.

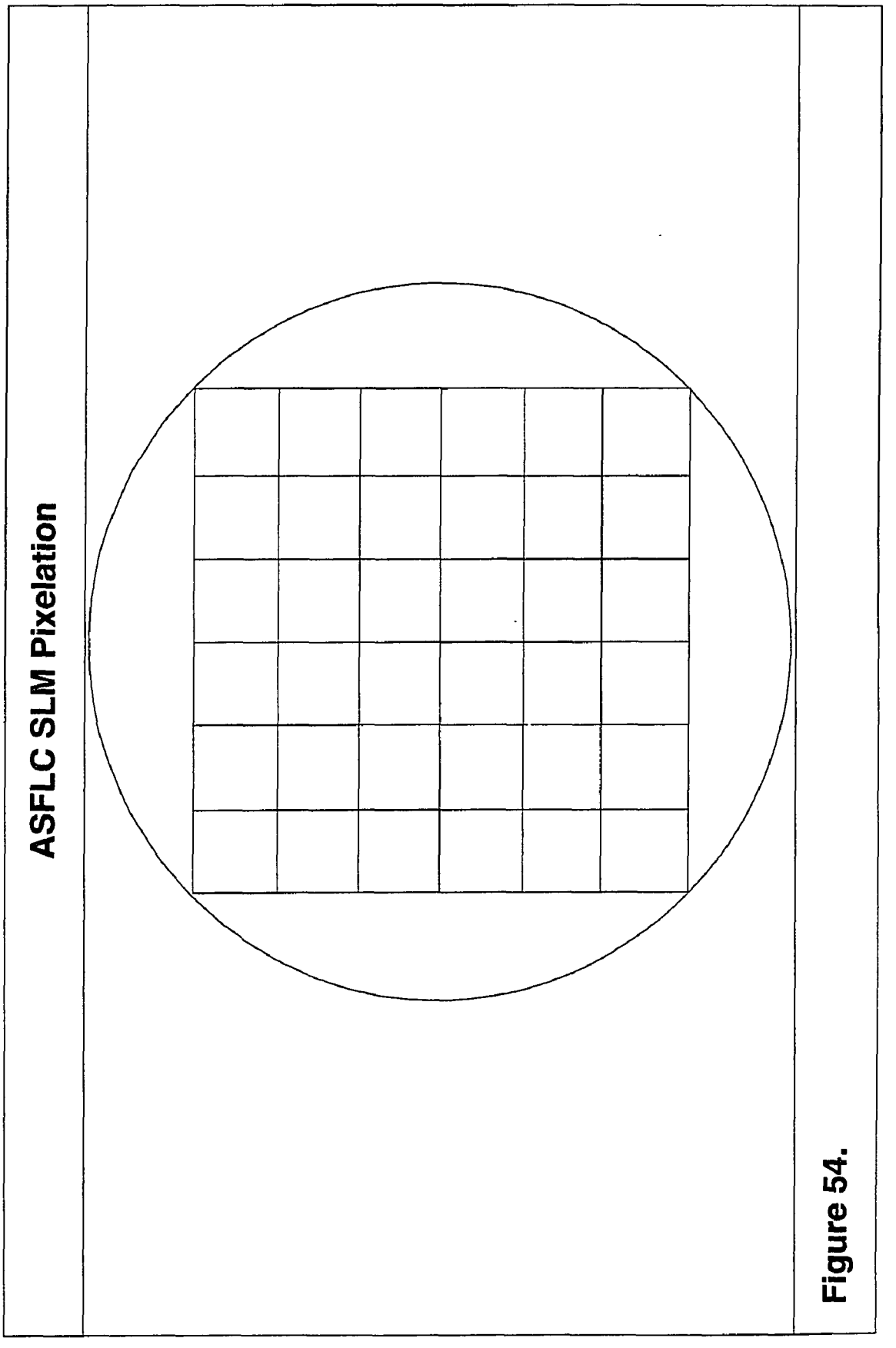

Figure 54.

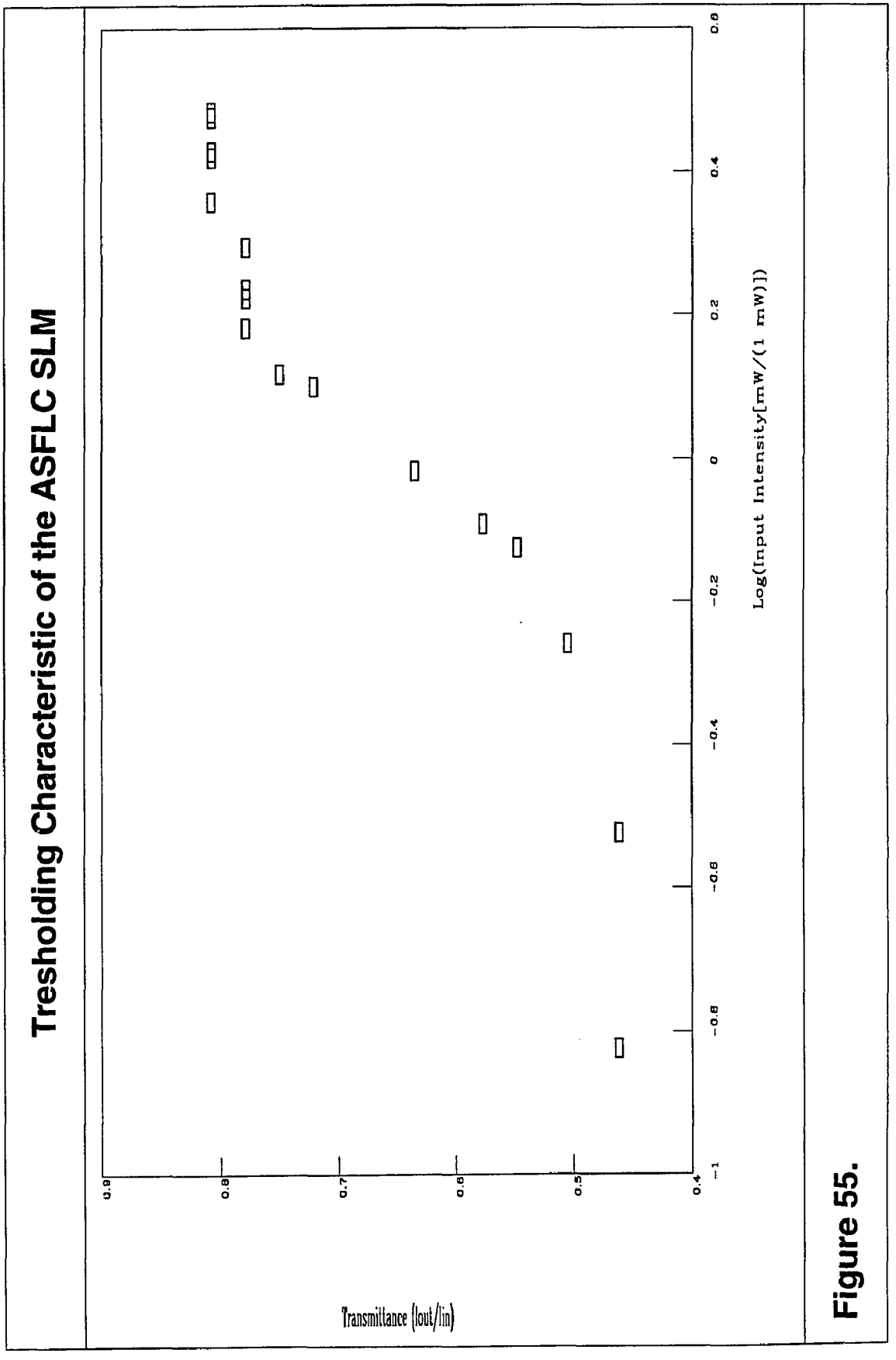
Tresholding Characteristic of the ASFLC SLM
Transmittance (Iout/Iin)
0.9
0.8
0.7
0.6
0.5
0.4
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
Log(Input Intensity[mW/(1 mW)])

Figure 55.

Comparison of Optically Addressed SLM's

	Aperture (mm)	Resolution (lp/mm)	Cycle Time (ms)
Displaytech (I)	10	163	0.07
Displaytech (II) *	27	200	0.05
Hoechst Celanese	40	38	25.00
Hughes LCLV	25	16	100.00
Hammamatsu MSLM	16	10	150.00

* Pending

Figure 56.

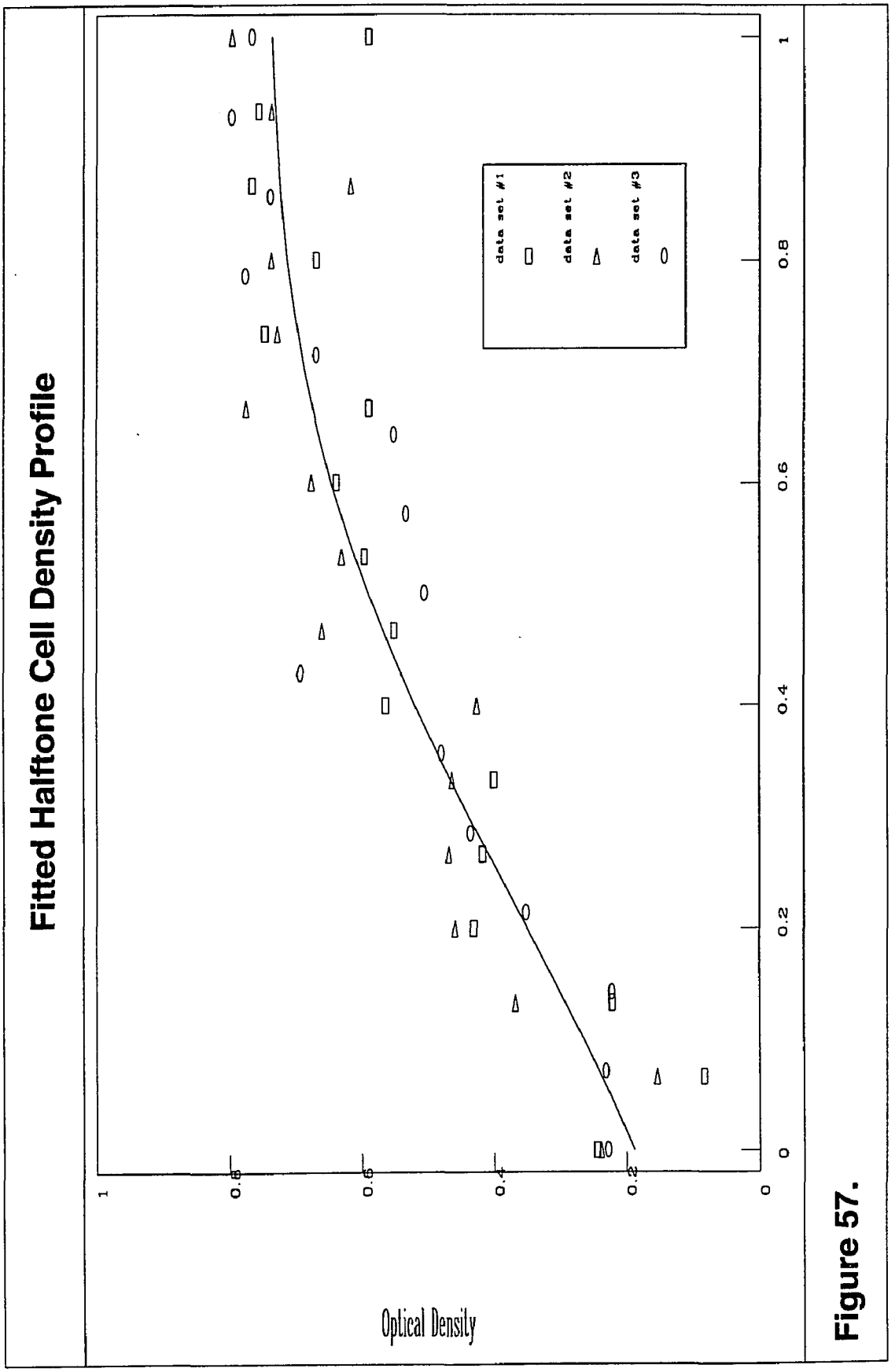
Fitted Halftone Cell Density Profile
data set #1
data set #2
data set #3
Optical Density
0
0.2
0.4
0.6
0.8
1

Figure 57.

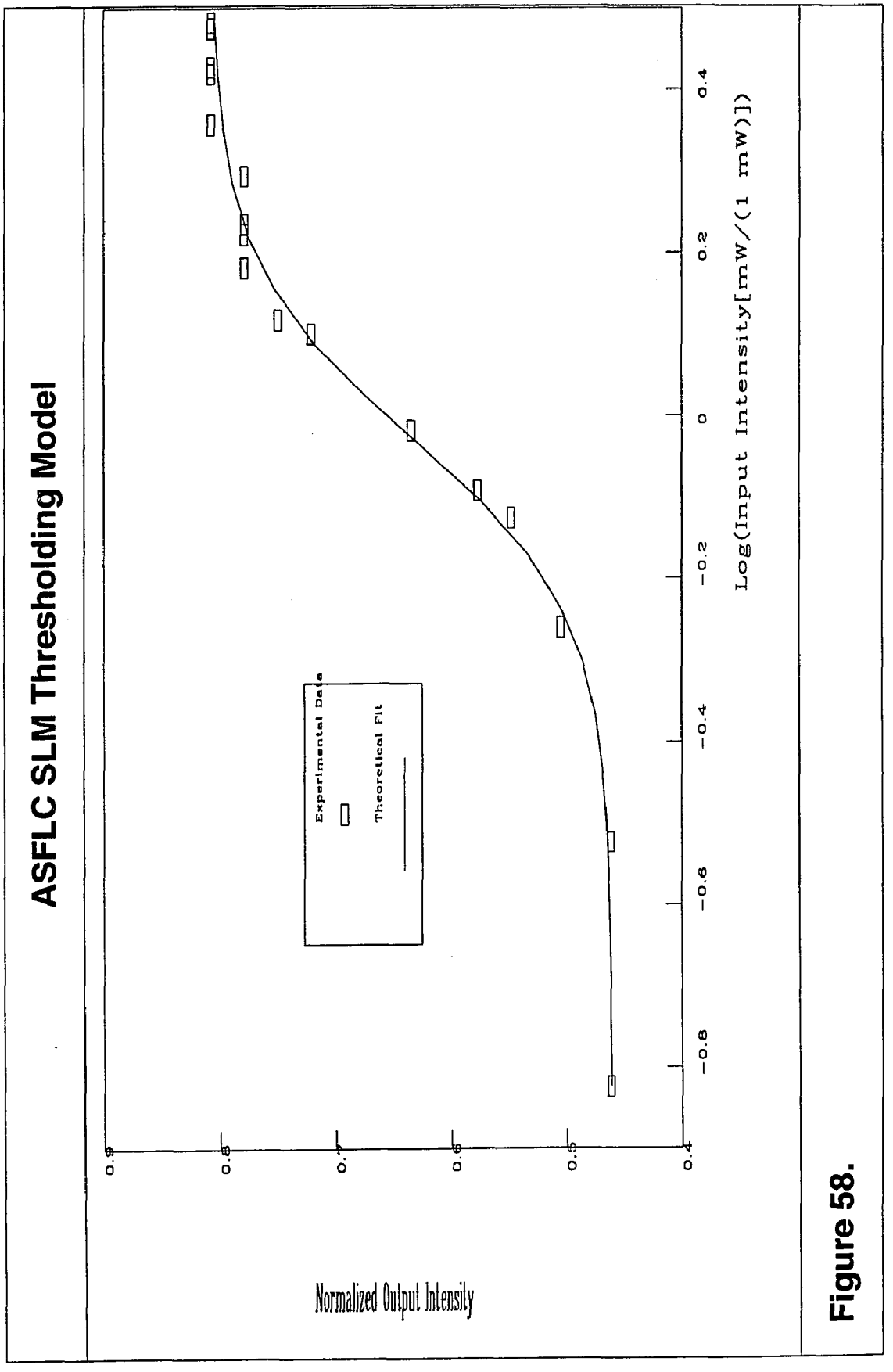
ASFLC SLM Thresholding Model
Normalized Output Intensity
Log(Input Intensity[mW/(1 mW)])
Experimental Data
Theoretical Fit
0.9
0.8
0.7
0.6
0.5
0.4
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8

Figure 58.

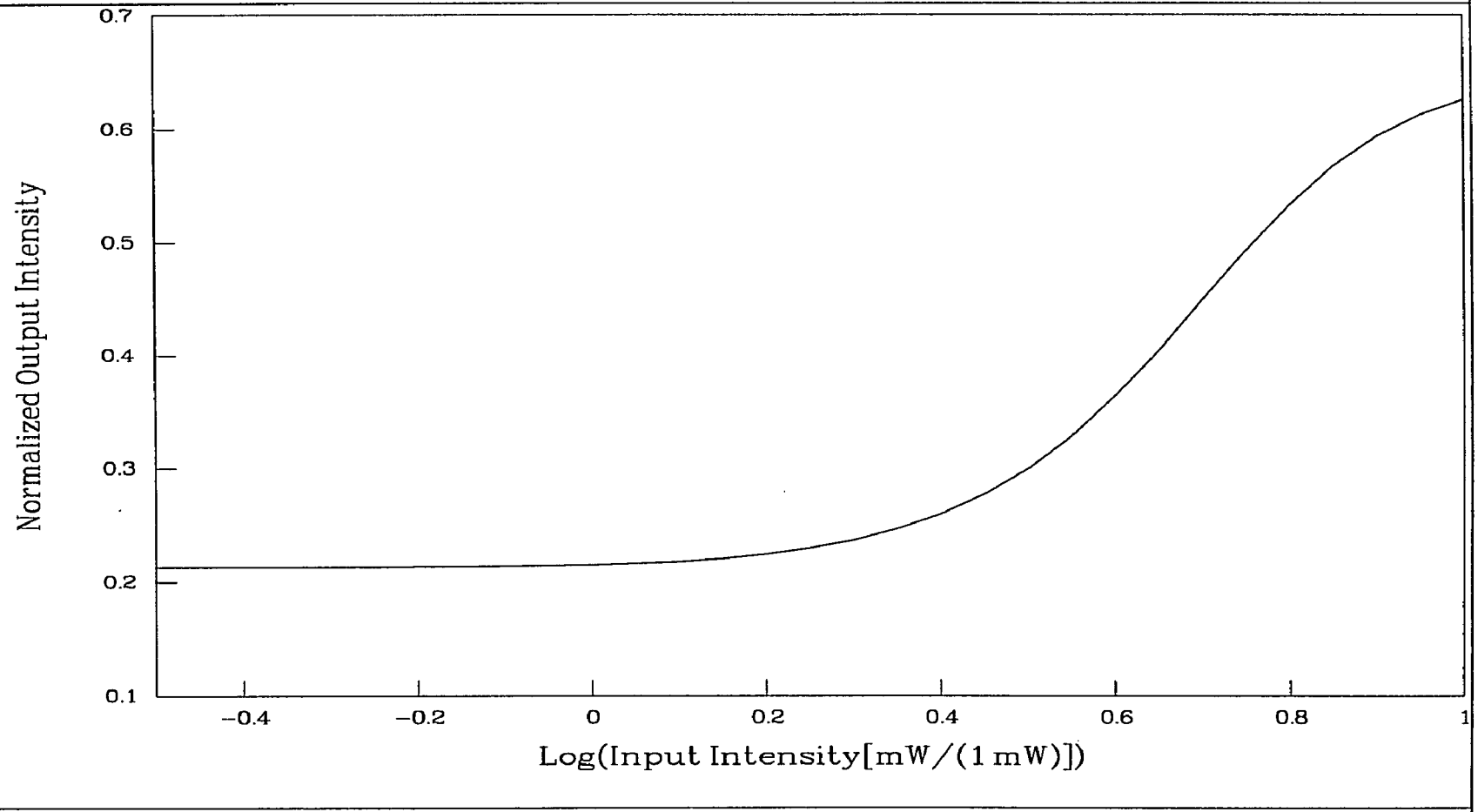
Predicted Transfer Function
Normalized Output Intensity
0.7
0.6
0.5
0.4
0.3
0.2
0.1
−0.4
−0.2
0
0.2
0.4
0.6
0.8
1
Log(Input Intensity[mW/(1 mW)])

Figure 59.

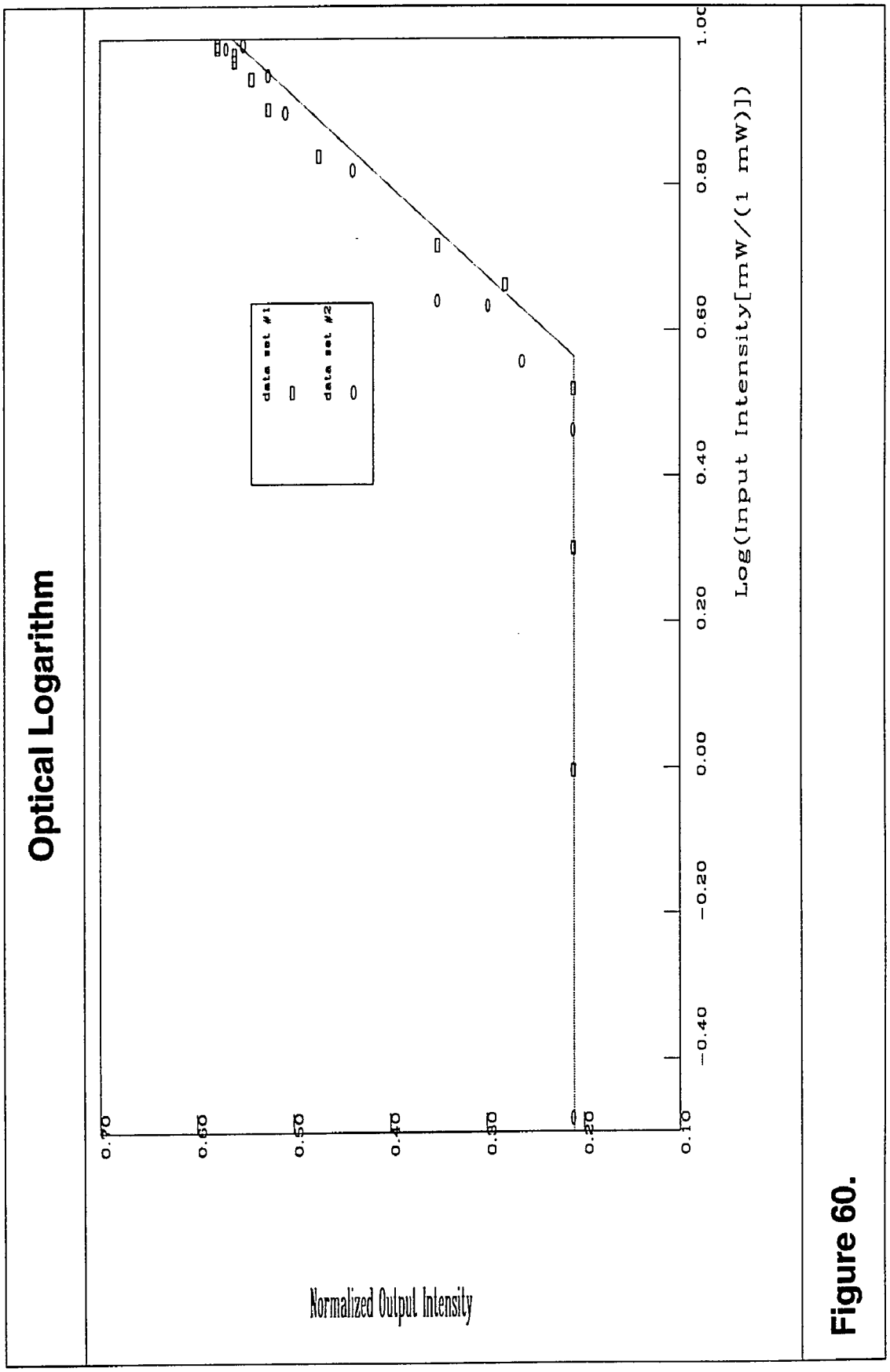
Optical Logarithm
data set #1
data set #2
Log(Input Intensity[mW/(1 mW)])
1.00
0.80
0.60
0.40
0.20
0.00
−0.20
−0.40
0.70
0.60
0.50
0.40
0.30
0.20
0.10
Normalized Output Intensity

Figure 60.

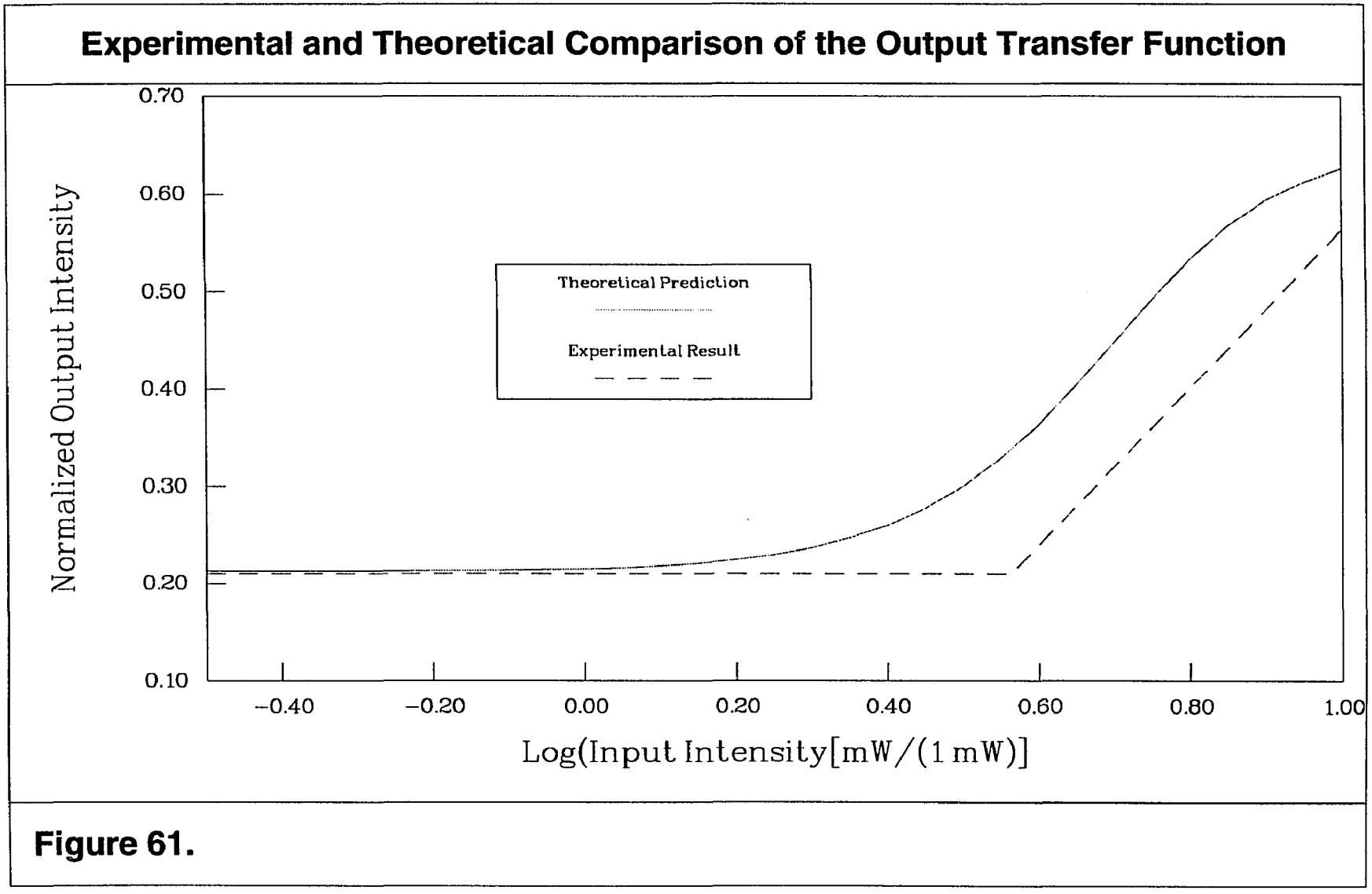
Experimental and Theoretical Comparison of the Output Transfer Function
Normalized Output Intensity
0.70
0.60
0.50
0.40
0.30
0.20
0.10
Theoretical Prediction
Experimental Result
−0.40
−0.20
0.00
0.20
0.40
0.60
0.80
1.00
Log(Input Intensity[mW/(1 mW)]

Figure 61.

4.3 2D Optical Iterative Processor

4.3.1 Introduction

An optical architecture based on optically addressed amorphous silicon ferroelectric liquid crystal spatial light modulators has been built and demonstrated for optical iterative processing of two dimensional images. The introduction of a two dimensional delay line within an optical ring allows control of the cycling rate of the optical iterative processor from a few Hertz to up to 15 kHz. Potential applications for real time optical iterative processing include scaling and rotation for optically based automated target recognition (ATR), optical fractal analysis, optical wavelet transform, and optical artificial neural network implementations.

An optical ring has previously been demonstrated[9]. Two major disadvantages prevent that implementation from being more useful: the optical gain for each iteration is less than unity, and the cycling rate is not controllable. Since the gain per cycle is less than unity, each cycle produces an optical power loss effectively limiting the number of usable cycles for information processing. The cycling is so fast that many succeeding processors may be unable to keep up with the data rate. A 2-D optical iterative processor can be designed with unity gain for each

cycle and a controllable cyclic rate by introducing a two dimensional delay line within an optical ring.

4.3.2 Optical Architecture Design

The optical architecture is comprised of two key subsections, an optical ring and an optical delay line.

There are many ways of implementing an optical ring. An optical ring is essentially a closed loop into which an injected optical pulse is introduced; the pulse cycles within the ring. The architecture that we are using is a rectangular ring (Figure 62).

To implement the 2-D optical delay line, two 2-D optically addressed spatial light modulators with memory must be used. The spatial light modulators, once they are introduced within the ring, must be able to read the information contained within the optical pulse cycling within the optical ring. They also must store the data until transferred from one spatial light modulator to the next spatial light modulator.

The optical modulators used are optically addressed hydrogenated amorphous silicon ferroelectric liquid crystal spatial light modulators (ASFLC

SLM) (Figure 63). The hydrogenated amorphous silicon layer acts as a photosensor. Light impinges upon it and produces a charge distribution proportional to the light intensity. This charge distribution modulates the ferroelectric liquid crystal layer, which then imposes the 2-D information onto an optical read beam. The ASFLC SLM has two electrically controlled states, write and erase. To implement the 2-D optical delay line three states are necessary: erase, write and read. One of the ASFLC SLMs is cleared of all previously held data (erased), data is written into it (the data input phase), the data is transferred to the other ASFLC SLM (the read phase), and then this cycle repeats (Figure 64). The write state of the SLM can be subdivided into a write with an optical pulse and write with no input data. The write with no input data produces no change in the data stored in the SLM and corresponds to the required read phase. The three SLM states are then obtained by modulating both the ASFLC SLM and the optical beam (Figure 65).

Next, the 2-D optical delay line is introduced within the optical ring. An electrically addressed SLM (a STC ferroelectric liquid crystal SLM) is used to inject data into the optical ring, and a camera is used to observe optical modulation within the optical ring (Figure 66). A series of pulse generators and amplifiers is used to ensure that the various states of the ASFLC SLMs are synchronized.

4.3.3 Results

Two experiments were carried out. First, the image of a circle was injected into the optical ring and allowed to cycle. When one of the legs of the optical ring was blocked the circle disappeared. In the other experiment, the image of a bar was introduced into the optical ring. At each cycle of the ring the bar was shifted downwards to determine the number of cycles the bar underwent and its distortion (Figure 67,68). The cycle rate used was 220 Hz.

4.3.4 Conclusions

Two problems became obvious. First, optical alignment is critical for a large number of cycles. The effects of any slight misalignment accumulate and became unacceptable. Second, the available ASFLC SLM devices are prototypes and have poor optical quality. This distorted the injected image with each pass.

Optical misalignments can be reduced by going to integrated optics. As the optically addressed ASFLC SLM or its equivalent becomes available the distortion problem should be reduced to acceptable levels.

This type of optical iterative processor, with dove prisms inserted to produce image rotation or zoom lenses to produce image scaling, can be used as a preprocessor for an optical matched filter for rotation and scale invariant pattern recognition.

The 2-D ASFLC SLMs have a sigmoid nonlinear transfer function making them ideal optical neurons. By adding connection weights, in the form of filters or holograms, a real time optical heteroassociative neural network can be implemented. Since the ASFLC SLMs have a resolution exceeding 80 lp/mm and a frame rate up to 15 kHz, a neural net with a large number of neural elements per layer and a large throughput can be optically implemented.

By introducing an electrically addressed SLM and a few minor modifications, an optical Wavelet transform can be implemented. Implementation of a 2-D wavelet transform is essentially a correlation, with respect to scale, of an input function with a wavelet function. The wavelet function can be scaled iteratively and correlated with the input function within the optical ring.

Other potential applications include, optical fractal analysis and texture analysis.

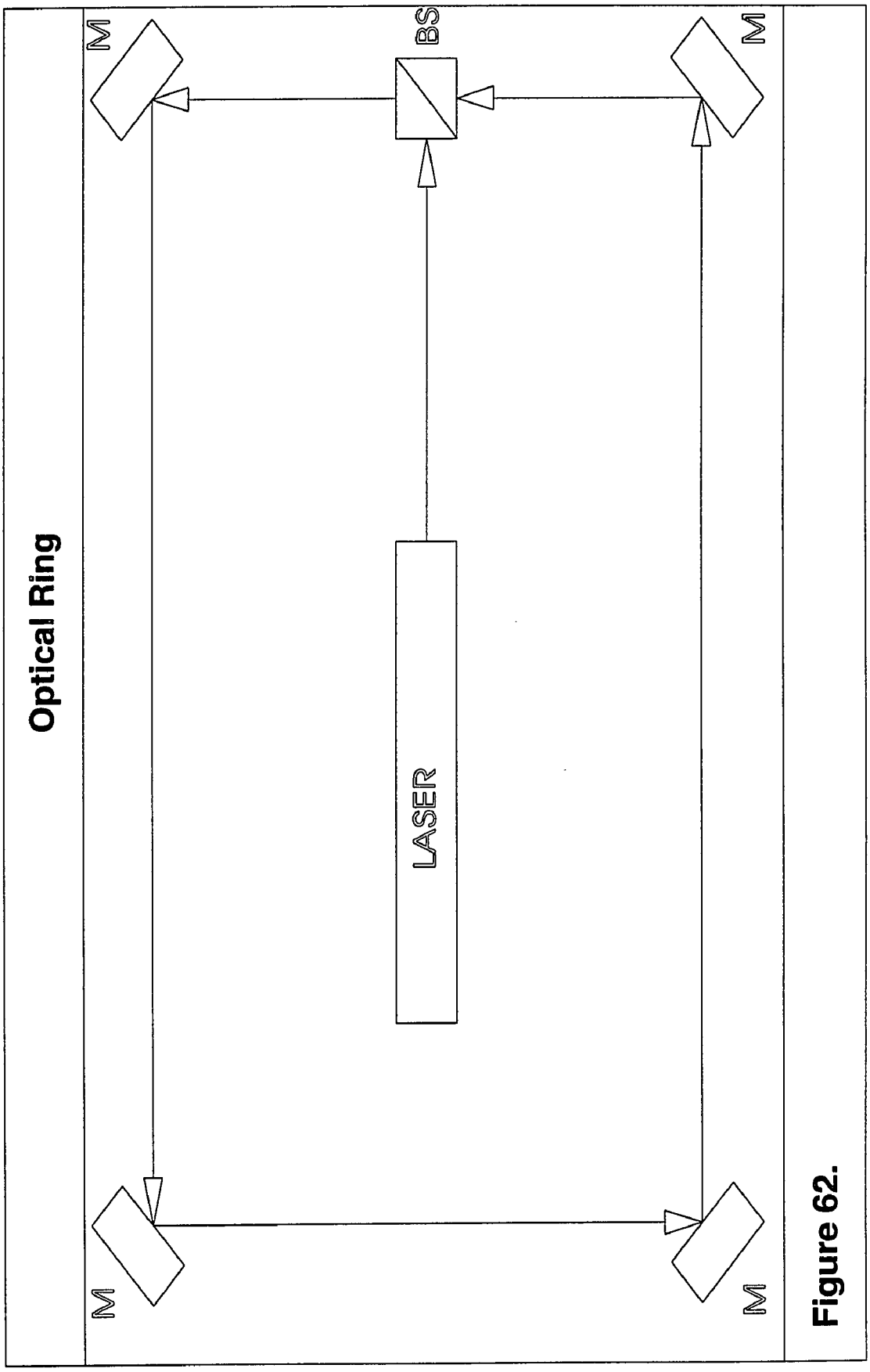

Figure 62.

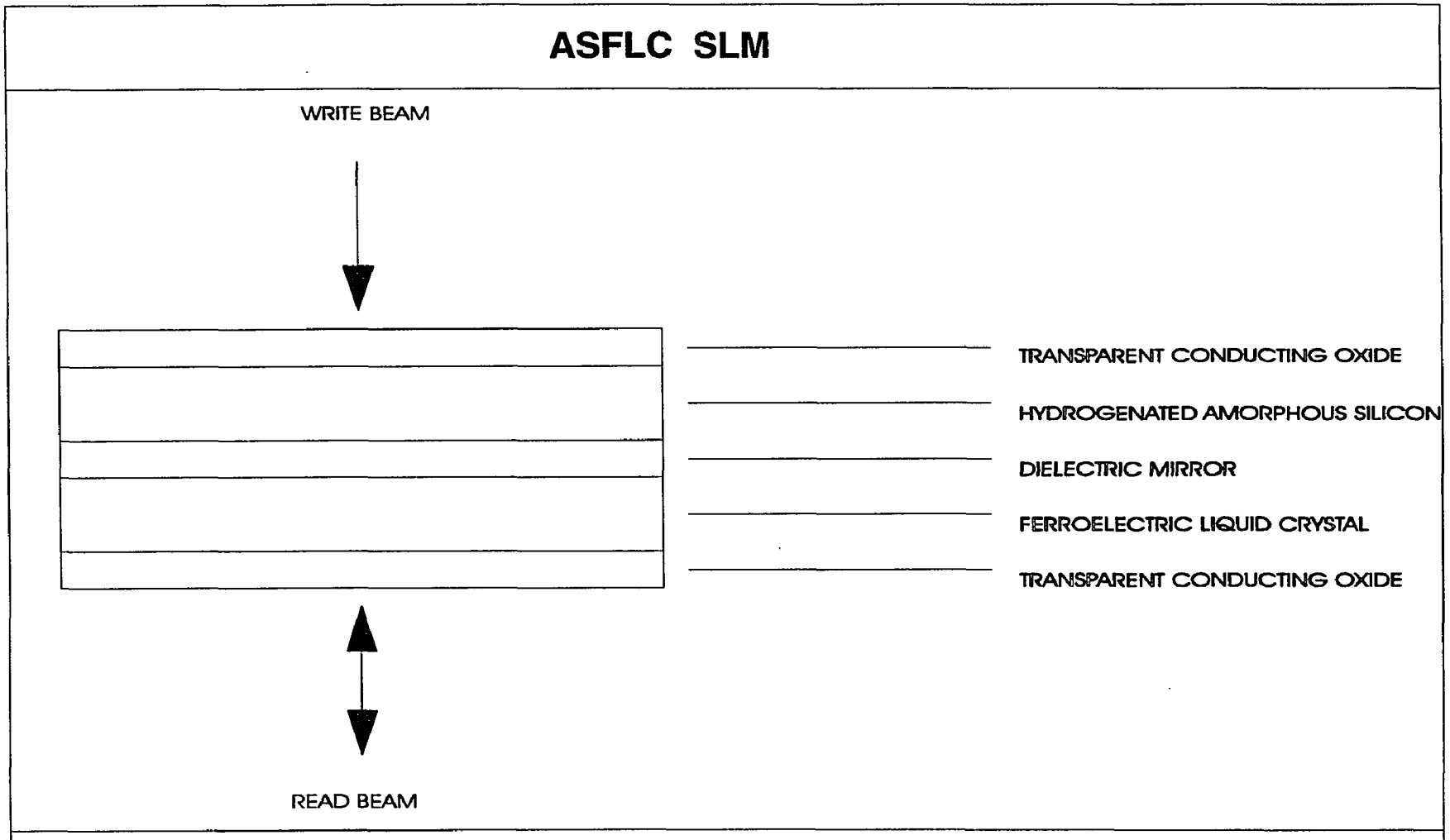
ASFLC SLM
WRITE BEAM
TRANSPARENT CONDUCTING OXIDE
HYDROGENATED AMORPHOUS SILICON
DIELECTRIC MIRROR
FERROELECTRIC LIQUID CRYSTAL
TRANSPARENT CONDUCTING OXIDE
READ BEAM

Figure 63.

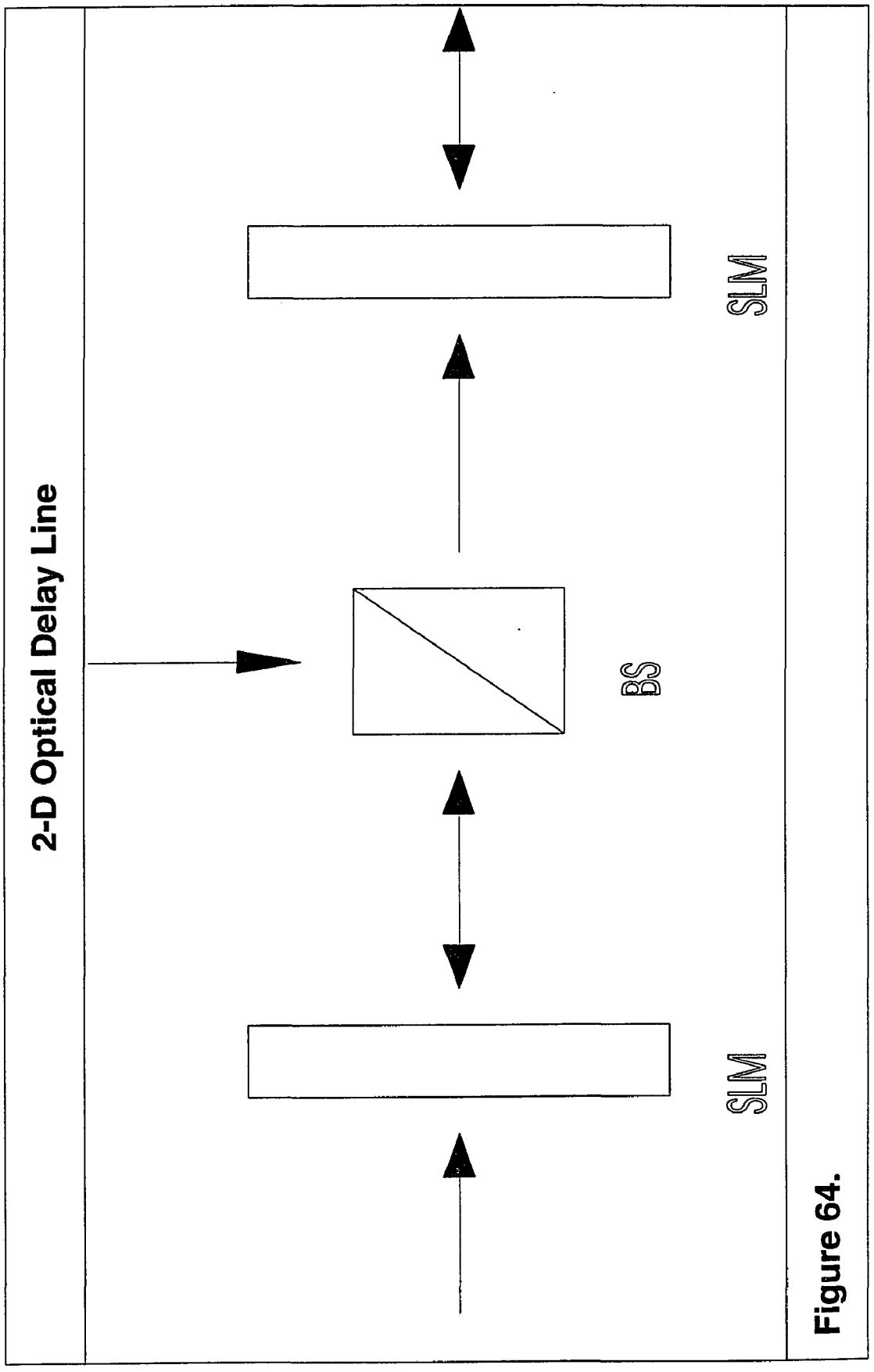

Figure 64.

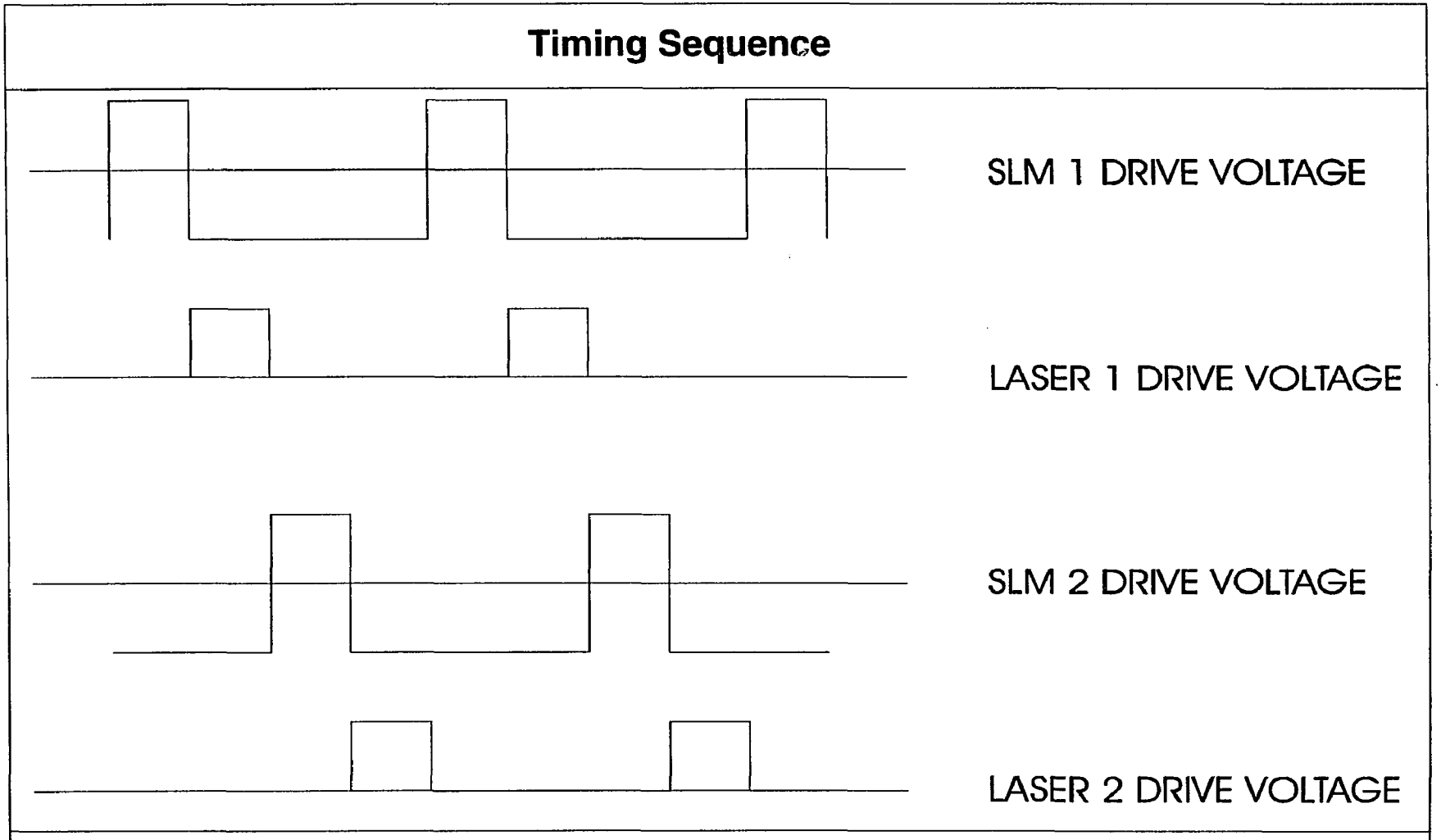
Timing Sequence
SLM 1 DRIVE VOLTAGE
LASER 1 DRIVE VOLTAGE
SLM 2 DRIVE VOLTAGE
LASER 2 DRIVE VOLTAGE

Figure 65.

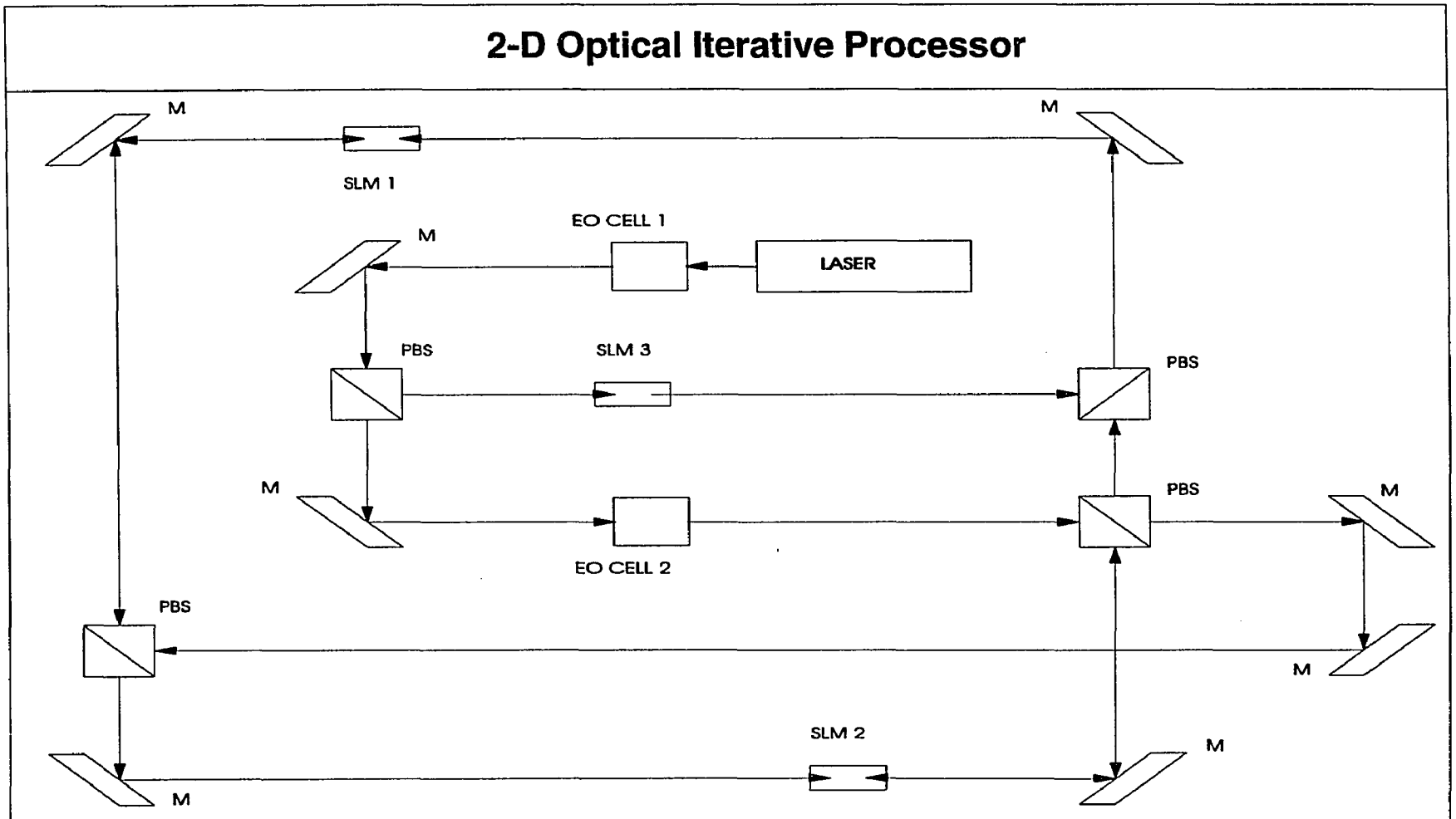
2-D Optical Iterative Processor
M
M
SLM 1
EO CELL 1
M
LASER
PBS
SLM 3
PBS
M
PBS
M
EO CELL 2
PBS
M
M
SLM 2
M
M

Figure 66.

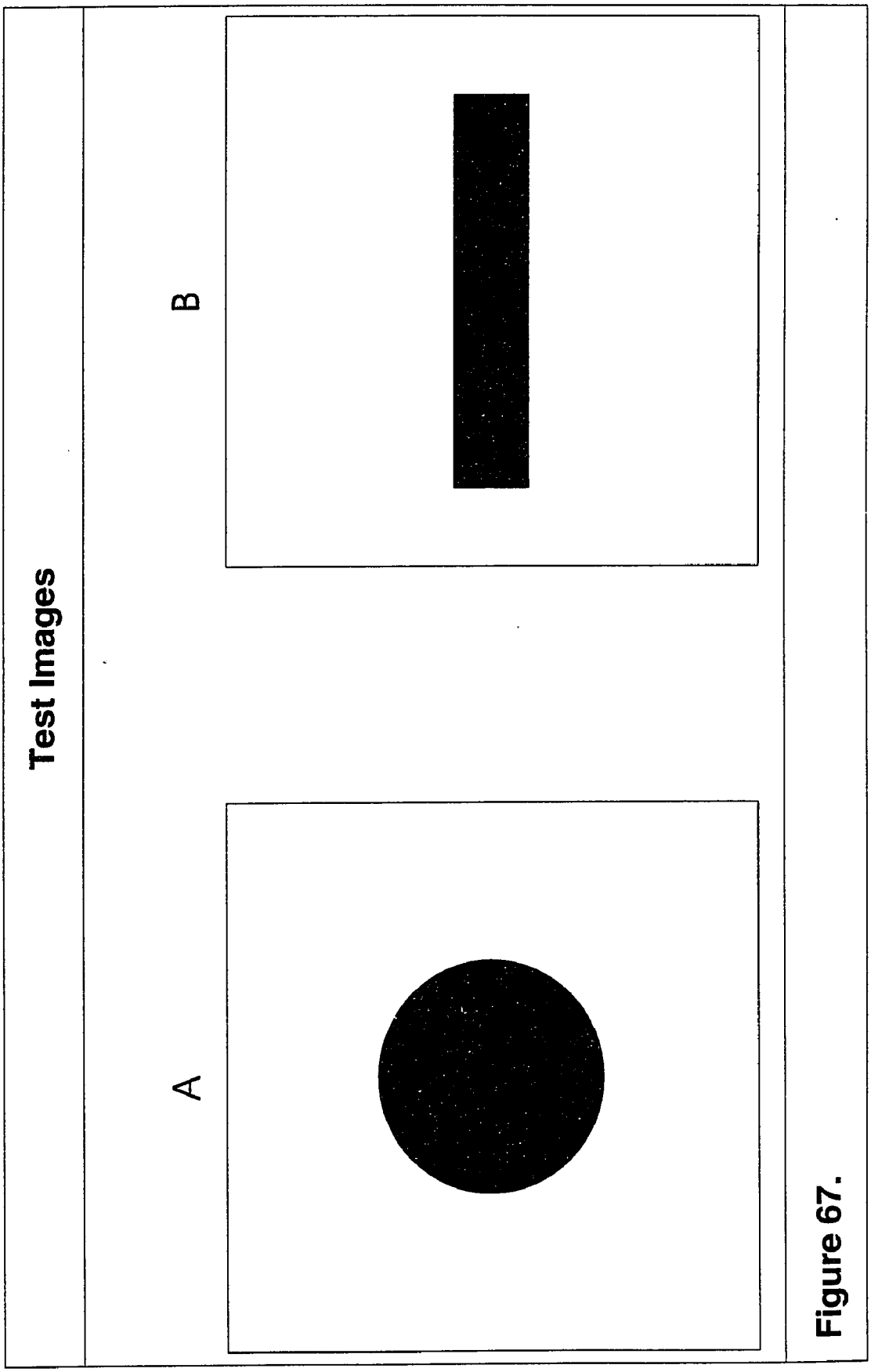

Figure 67.

2-D Optical Iterative Processor Output

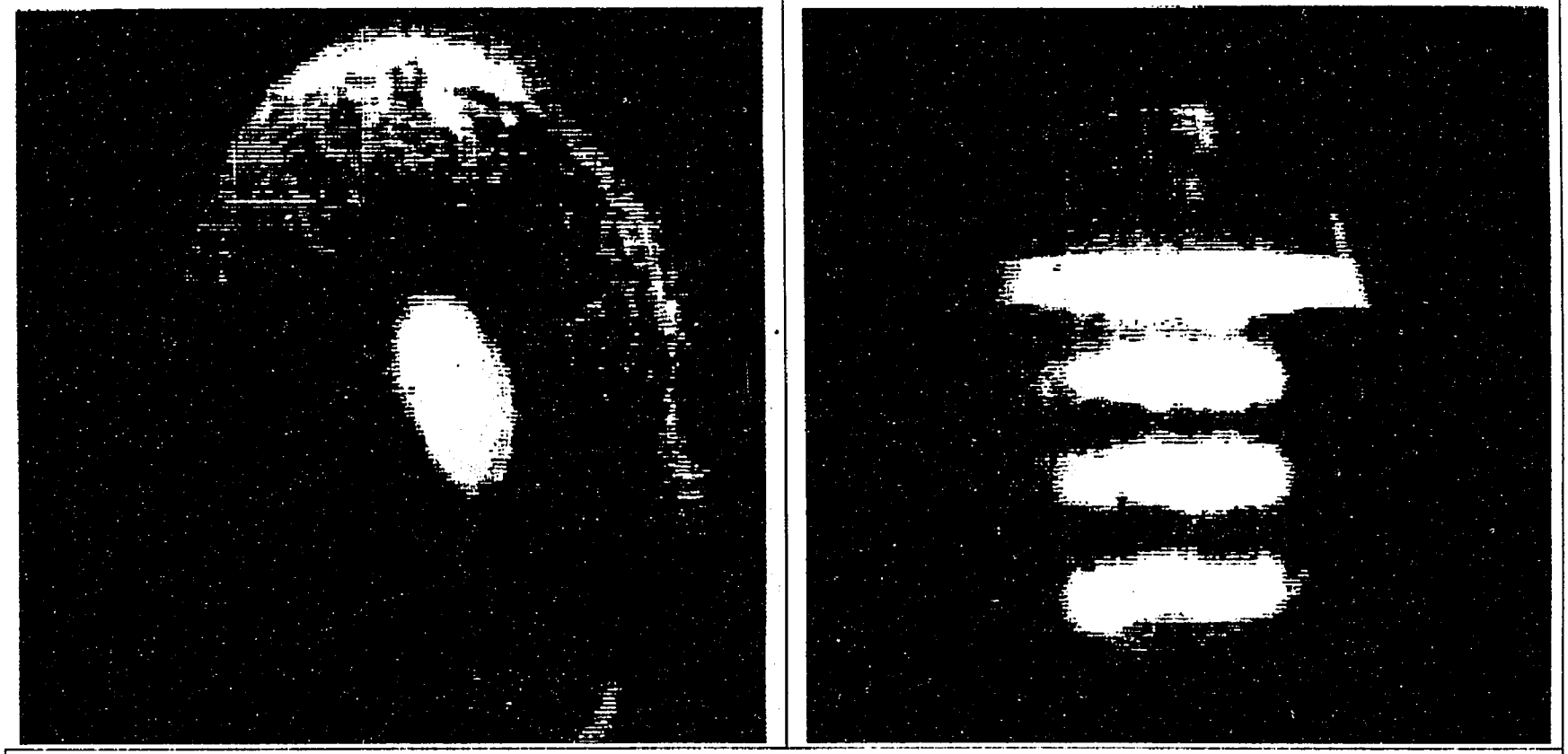

Figure 68.

5 Conclusions

It was shown in Section 3.2 that a technique called N-Wavelet Coding can differentiate highly correlated objects, and that by increasing the number N of features it can provide increased noise tolerance in the pattern detection and classification process[59] .

In Section 3.3 it was shown, that if it is assumed that complex fractal texture are a superposition of simpler fractal basis textures, a unique characterization of the fractal textures can be obtained with a vector comprised of the estimated fractal dimensions of each of the discriminating fractal basis textures. The technique demonstrated in Section 3.3, can be used to extract the fractal basis textures which allows discrimination between complex basis textures[60].

In Section 3.4 it is shown that using a fractal scanning technique it is possible to retain sufficient adjacency information to enable detection and classification of 2-D objects from a 1-D vector obtained through a fractal sampling technique, in this case a Peano space filling curve[61].

In Section 4.2 it is shown that a grey level optical processing can be obtained from a binary optical device by using a halftoning technique[62].

In Section 4.3 an optical iterative processor capable of implementing a wavelet transform using time multiplexed technique is demonstrated[63].

In conclusion a new technique for uniquely characterizing complex fractal texture comprised of both deterministic and stochastic feature has been developed and demonstrated. A method for optically implementing the technique has been developed and key subsystem demonstrated. Various parts of this thesis has been published in open literature[59-63], published as a master's thesis[32], and issued as a patent[64].

6 Thesis Related Publications

The thesis related publications are listed below:

S. Phuvan, "Optical Implementation of N-Wavelet Coding for Pattern Classification," App. Optics, Special feature "Applications of Limited Extent Waves", Oct. 94

S. Phuvan, "Texture Characterization," Invited Paper, World Congress on Artificial Neural Networks, June 1994

S. Phuvan, H. Szu, N. Caviris, T. Oh, and L. Welsh, "Texture analysis by space-filling curves and 1-D Haar wavelets," Optical Engineering, vol. 31, no. 9, pp. 1899, September 1992

S. Phuvan, N. Caviris, J. Gray, and T. Rice, "Real time optical logarithm using an ASFLC SLM," Proceedings of the SPIE, April 1992

S. Phuvan and V. Israni, "2-D Optical iterative processor," Proceedings of the SPIE, April 1993

Sole inventor: S. Phuvan, "Variable Rate Transfer of Optical Information," Patent no. 5,289,304 (1994), Assignee: United States of America

Vivek Israni, "Ferroelectric Liquid Crystal based Optical Ring Rotator Image Preprocessor for Automatic Pattern Recognition," MIT Master's Thesis, September 1991, Thesis Supervisor: S. Phuvan, Thesis Advisor: Dr. Cardinal Warde

7 Acknowledgements

I would like to acknowledge the support of NSWC without which the work presented could not be completed. I am deeply indebted to Nick Caviris, Mary Lacey, friends and family for their help, knowledge and moral support throughout the course of this work. I owe special thanks to Dr. Yao Li for his encouragement, to Roger Israni who toiled long and hard under my supervision on the optical iterative processing section, Dr. Harold Szu and Dr. Tae Kwon Oh for their colloborations for the work presented in the fractal scanning section, John Grey and Dr. Theodore Rice for their colloborations for the work presented in the real time halftoning section.

8 Bibliography

1 . Y. Abu-Mustafa and D. Psaltis,"Optical Neural Computers," Sci. Am. vol. 256, pp. 88-95, 1987

2 . Ahmad Armand, "Real Time Nonlinear Optical Information Processing," PhD Thesis, University of Southern California (June 1979)

3 . J.Aloimonos and M.J. Swain, "Shape from texture," Int. Joint. Conf. Artificial Intelligence, pp. 926-931, August 1985

4 . J. Aloimonos and M.J. Swain, "Shape from patterns: Regularization," Int. J. Computer Vision, vol. 2, pp. 171-187, 1988

5 . R. Badii and A. Poliki, "Statistical description of chaotic attractors: the dimension function," J. Stat. Phys. 40, 725 (1985)

6 . J.J. Bartholdi and L.K. Platzman, "Heuristics based on Space-filling Curves for Combinatorial Problems in the Plane," PDRC Technical Report 84-08, Georgia Institute of Technology, 1984

7 . B.L. Bullock, "Finding Structure in Outdoor-Scenes," Pattern Recognition and Artificial Intelligence, C.H. Chen, ed., (North-Holland, Amsterdam, 1976), pp. 61-85

8 . M.J. Carlotto and M.C. Stein, "Detecting Man-Made Change in Imagery," Intelligent Robots and Computer Vision: Seventh in a Series, D.P. Casasent, ed., Proc. Soc. Photo-Opt. Instrum. Eng., 1002, pp. 6-11 (1985)

9 . Arthur E. T. Chiou and Pochi Yeh, "Scaling and Rotation of Optical Images Using a Ring Cavity," Applied Optics, vol. 29, No. 11, pp. 1584-1586 (10 April 1990)

10 . C.K. Chui, "An Introduction to Wavelets," Academic Press, New York, 1992

11 . G.R. Cross and A.K. Jain, "Markov random field texture models," IEEE Trans. PAMI, Vol. PAMI-5, no. 1, p.25-39 (Jan. 1983)

12 . S.R. Dashiell and A.A. Sawchuck, "Nonlinear optical processing: analysis and synthesis," Applied Optics, vol. 16, no. 4, 1009-25 (April 1977)

13 . S.R. Dashiell and A.A. Sawchuck, "Nonlinear Optical Processing: nonmonotonic halftone cells and phase halftones," Applied Optics, vol. 16, no. 7, 1936-43 (July 1977)

14 . S.R. Dashiell and A.A. Sawchuck, "Nonlinear optical processing: effects of input medium and precompensation," Applied Optics, vol. 16, no. 8, 2279-2287 (August 1977)

15 . R. Duda and P. Hart, Pattern Classification and Scene Analysis, John Wiley & Sons, New York, 1973

16 . D.J. Elliott, "Integrated Circuit Mask Technology," New York, McGraw-Hill, 1964

17 . K.J. Falconer, "The Geometry of Fractal Sets, " vol. 85 of Cambridge Tracts in Mathematics (Cambridge Univ. Press, Cambridge 1985)

18 . J.D. Farmer, E. Ott, and J.P. Yorke, "The Dimension of Chaotic Attractors," Physica 7D, 153 (1983)

19 . W. Fong, T. Cline, M. Walker, and S. Rosenberg, "A photolithographic advisor: an expert system," Proc. SEMICON/EAST 85, pp. 1-8, 1985

20 . E. Freysz, B. Pouligny, F. Argoul and A. Arneodo, "Optical Wavelet Transform of Fractal Aggregate," Preprint., May 9, 1989

21 . P. Grassberger and I. Procaccia, "Measuring the strangeness of strange attractors," Physica 9D, 189 (1983)

22 . P. Grassberger and I. Procaccia, "Characterization of Strange Attractors," Phys. Rev. Lett., 50, 346 (1983)

23 . P. Grassberger, "Generalizations of the Hausdorff dimension of fractal measures," Phys. Lett. A 107, 101 (1985)

24 . P. Grassberger, R. Radii, and A. Politi, "Scaling laws for invariant measures on hyperbolic and non-hyperbolic attractors," J. Stat. Phys. 51, 135 (1988)

25 . H.S. Greeside, A. Wolf, J. Swift, and T. Pignataro, "Impracticality of a box counting algorithm for calculating the dimensionality of strange attractors," Phys. Rev. A 25, 3453 (1982)

26 . A.D. Gross and A. Rosenfeld, "Multiresolution Object Detection and Delineation," Computer Vision Graphics Image Processing 39, 102-115 (1987)

27 . R.M. Haralick, "Statistical and structural approaches to texture," Proceedings of the IEEE, vol. 67, pp. 786-804, May 1979

28 . F. Hausdorff, "Dimension und Ausseres Mass," Math. Annalen 79, 157 (1919)

29 . J. Helman and L. Hesselink, "Representation and display of vector field topology in fluid flow data sets," Computer, pp. 27-36, August 1989

30 . Vivek Israni, "Ferroelectric Liquid Crystal based Optical Ring Rotator Image Preprocessor for Automatic Pattern Recognition," MIT Master's Thesis, September 1991, Thesis Supervisor: S. Phuvan, Thesis Advisor: Dr. Cardinal Warde

31 . B. Julesz, "Texton gradients: the texton theory revisited," Biol. Cybernetics, vol. 54, no. 4-5, p.245-51 (1986)

32 . R.L. Kashyap and A. Khotanzad, "A model-based method for rotation invariant texture classification," IEEE Trans. PAMI, vol. PAMI-8, no. 4, p.472-481 (July 1986)

33 . M. Kass and A. Witkin, "Analyzing oriented patterns," Computer Vision, Graphics and Image Processing, vol. 37, pp. 362-385, 1987

34 . Hisatoyo Kato and J.W. Goodman, "Nonlinear Transformations and Logarithmic Filtering in Coherent Optical Systems," Optics Communications, vol. 8, no. 4, 378-381 (August 1973)

35 . J.M. Keller, S. Chen, and R.M. Crownover, "Texture Description and Segmentation through Fractal Geometry," Computer Vision, Graphics, and Image Processing, no. 45, pp. 150-166, 1989

36 . A.N. Kolmogorov, "A New Invariant for Transitive Dynamical Systems," Dokl. Akad. Nauk SSSR 119, pp. 861-864 (1958)

37 . Lee and Moon, "An Optical Technique for Measuring Fractal Dimension," Phys. Lett., 114A, p.222 (1986)

38 . S.Y. Lu and K.S. Fu, "A syntactic approach to texture analysis," Comp. Graph. and Image Proc., vol. 7, pp. 303-330, 1978

39 . R.P. Lippman, "An introduction to computing with neural networks," IEEE ASSP, pp. 4-22, April 1987

40 . T. Lundahl, W. Ohley, S.K. Kay and R. Siffert, "Fractional Brownian Motion: a Maximum Likehood Estimator and Its Application to Image Texture," IEEE Trans. Med. Imaging, **MI-5**, pp. 152-161 (1986)

41 . W.S. MacKenzie, "Atlas of Igneous Roscks and Their Textures," John Wiley, New York, 1982

42 . B.B. Mandelbrot and B.J. Van Ness, "Fractional Brownian Motion, Fractional Noises and Applications," SIAM Rev. 10, pp. 442-438 (1968)

43 . B.B. Mandelbrot, "The Fractal Geometry of Nature," Freeman, New York, 1983

44 . B. Mandelbrot, "Self affine fractal sets, I: the basic fractal dimensions," Fractals in Physics, L. Pietronero and E. Tosatti, eds., Elsevier, Amsterdam 1986, pp. 3-15

45 . Steve Matsumoto and Bede Liu, "Halftone screen design methods for nonlinear optical processing," Applied Optics, vol. 18, no. 16, 2792-2802, 15 August 1979

46 . G.G. Medioni, and Y. Yasumoto, "A Note for Using Fractal Dimension for Segmentation," Proc. of the 1984 Workshop of Computer Vision, Representation and Control, Annapolis, MD, April 30th-May 2nd, 1984, pp. 25-30

47 . B. Moghaddam, K.J. Hintz, C.V. Stewart, "Dimension and Lacunarity Measurements of IR Images Using Hilbert Scanning," SPIE's OE/Aerospace Sensing Proceedings, April 2, 1991

48 . P.S. Mohariar, "Deconvolution by Cepstral Transformation," Journal of the Institution of Electronics and Telecommunication Engineers, vol. 28, no. 10, Oct 1982

49 . J.P. Murray, "The trade-offs in neural net implementations," High Performance Systems, pp. 74-78, March 1990

50 . R. Nevatia and K.E. Price, "Locating Structure in Aerial Images," Digital Image Analysis, Vol. 2 of Digital Image Processing and Analysis, R. Chllapa and A.A. Sawchuck, eds., (IEEE, New York, 1985), pp. 247-255

51 . P.T. Nguyen and J. Quinqueton, "Space Filling Curves and Texture Analysis," 6th Int'l Conf. on Pattern Recognition, Munich, Germany, Oct. 19-22, 1982, pp. 282-285.

52 . A.V. Oppenheim and R.W. Schafer, "Digital Signal Processing," Prentice-Hall, Englewood Cliffs, 1975

53 . G. Paladin and A. Vulpiani, "Anomalous Scaling Laws in Multifractal Objects," Phys. Rep. 156, 147 (1987)

54 . G. Peano, "Sur une Courbe qui Remplit Toute en Aire Plaine," Math. Ann., Vol. 36, 1890

55 . S. Peleg, J. Naor, R. Hartely and D. Avnir, "Multiple Resolution Texture Analysis and Classification," IEEE Trans. Pattern Anal. Mach. Intell. PAMI-6, pp. 518-523 (1984)

56 . S. Peleg, J. Naor, R. Hartely and D. Avnir, "Multiple Resolution Texture Analysis and Classification," IEEE Trans. Pattern Anal. Mach. Intell. PAMI-6, pp. 518-523 (1984)

57 . T. Peli, "Multiscale fractal theory and object recognition," J. Opt. Soc. Am. A, vol. 7, no. 6, June 1990

58 . A. Pentland, "Fractal Description of Natural Scenes," IEEE Trans. Pat. Ana. and Machine Intelligence, vol. 6, pp. 661-674, Nov. 1984

59 . S. Phuvan, "Optical Implementation of N-Wavelet Coding for Pattern Classification," App. Optics, Special feature "Applications of Limited Extent Waves", Oct. 94

60 . S. Phuvan, "Texture Characterization," Invited Paper, World Congress on Artificial Neural Networks, June 1994

61 . S. Phuvan, H. Szu, N. Caviris, T. Oh, and L. Welsh, "Texture analysis by space-filling curves and 1-D Haar wavelets," Optical Engineering, vol. 31, no. 9, pp. 1899, September 1992

62 . S. Phuvan, N. Caviris, J. Gray, and T. Rice, "Real time optical logarithm using an ASFLC SLM," Proceedings of the SPIE, April 1992

63 . S. Phuvan and V. Israni, "2-D Optical iterative processor," Proceedings of the SPIE, April 1993

64 . Sole inventor: S. Phuvan, "Variable Rate Transfer of Optical Information," Patent no. 5,289,304 (1994), Assignee: United States of America

65 . K. Prazdny, "On the information in optical flows," Computer Vision, Graphics and Image Processing, vol. 22, pp. 139-159, 1983

66 . D. Psaltis, D. Brady, K. Wagner, "Adaptive Optical Network using Photorefractive Crystals," Appl. Opt., vol. 27, pp. 1752-1759, 1988

67 . T.F. Quatieri, "Object Detection by Two Dimensional Linear Prediction," Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (IEEE, New York, 1983), pp. 108-111

68 . A. Renyi, "Probability Theory," North-Holland, Amsterdam (1970)

69 . O. Rioul and M. Vetterli, "Wavelets and Signal Processing," IEEE Signal Processing, pp. 14-38, Oct. 91

70 . Samat, "Computer Image Storage," University of Maryland, 1988

71 . Patrick K. Simpson, "Artificial Neural Systems," Pergamon Press, Elmsford, 1990

72 . J. Solka, C. Priebe, and G. Rogers , "An Initial Assessment of Discriminant Surface Complexity for Power Law Features Using Adaptive Mixture Neural Networks," NAVSWC, 1991

73 . H. Szu, "Reconfigurable neural nets by Energy Convergence Learning Principle based on extended McCulloch and Pitts Neurons and Synapses," Int. Joint Conf. Neural Network-89, p. I-485, Washington D.C. June 18-22, 1989

74 . H. Szu, "Neural Networks Based on Peano Curves and Hairy Neurons," Telematics & Informatics, Vol. 7, pp. 403-430, 1990

75 . H. Szu & K. Scheff, "Simulated Annealing Feature Extraction from Occluded and Cluttered Objects," Int. Joint Conf. Neural Networks-90, p. II-76, Washington D.C. Jan 15-18, 1990

76 . H. Szu, "Color Noise Annealing Benchmark by Exhaustive Solution of Travelling Salesman Problem," Int. Joint Conf. Neural Networks-90, p. I-317, Washington D.C. Jan. 15-18, 1990

77 . H. Szu, Yunlong Sheng and Jing Chen, "The Wavelet Transform as a Bank of Matched Filters," to be published in Appl. Opt., 1992

78 . F. Takens, "Invariants related to dimension and entropy," Atas do (Coloqkio Brasiliero do Matemtica, Rio de Janeiro, 1983)

79 . F. Tomita, Y. Shirai, and S. Tsuji, "Description of textures by a structural analysis," IEEE Trans. Pat. Analysis and Machine Intelligence, vol. 4-(2), pp. 183-191, 1982

80 . J. T. Tou, R. C. Gonzalez, "Pattern Recognition Principles," Addison-Wesley, Reading, 1974

81 . H. Voorhees, "Finding texture boundaries in images," MIT AI lab, Technical Report 968, June 1987

82 . H. Voorhees and T. Poggio, "Detecting textons and texture boundaries in natural images," First International Conference on Computer Vision, pp. 250-258, 1987

83 . R.F. Voos, "Random fractals: characterization and measurement," Scaling Phenomena in Disordered Systems, R. Pymm and A. Skjeltrop, eds., Plenum, New York 1985, pp. 1-11

84 . R.F. Voos, "Random fractal forgeries," Algorithms for Computer Graphics, Vol. F17 of NATO ASI Series, R.A. Earnshaw, ed., Springer-Verlag, Berlin 1985, pp. 805-835

85 . H. Wechsler, "Texture Analysis: A Survey," Signal Processing, Vol. 2, p. 271-282, 1980

86 . N. Yokoya, K. Yamamoto, and N. Funakubo, "Fractal -based analysis and interpolation of 3d natural surface shapes and their application to terrain modeling," CVGIP, vol. 43, no. 3, pp. 284-302, June1989

87 . Yan Zhang and Yao Li, "Opto-electronic Gabor Detector for Transient Signals," SPIE Conference in Orlando, Florida, February 1991

88 . Yan Zhang and Yao Li, "Optical Gabor and wavelet expansions of one and two dimensional signals," Proceedings of the SPIE, vol. 1702, p.20-40, (1992)