

(12) PATENT
(19) AUSTRALIAN PATENT OFFICE

(11) Application No. **AU 199655287 B2**
(10) Patent No. **700265**

(54) Title
Method and system for representing a data set with a data transforming function and data mask

(51)⁶ International Patent Classification(s)
H04N 001/41 **H04N** 007/30
G06T 009/00

(21) Application No: 199655287 (22) Application Date: 1996 .03 .27

(87) WIPO No: W096/31975

(30) Priority Data

(31) Number	(32) Date	(33) Country
08/415559	1995 .04 .03	US

(43) Publication Date : 1996 .10 .23
(43) Publication Journal Date : 1996 .12 .12
(44) Accepted Journal Date : 1998 .12 .24

(71) Applicant(s)
Iterated Systems, Inc.

(72) Inventor(s)
Barnsley Michael F.

(74) Agent/Attorney
SPRUSON FERGUSON

(56) Related Art
US 4941193
US 5065447



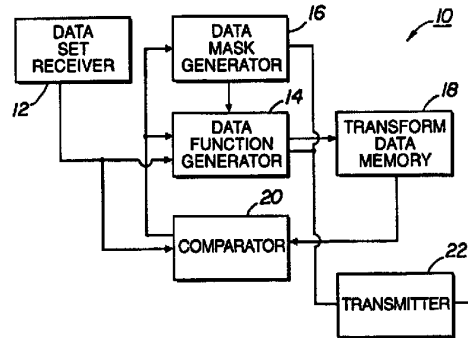
INT.

(51) International Patent Classification 6 : H04N 1/41, G06T 9/00, H04N 7/30		(11) International Publication Number: WO 96/31975
A1		(43) International Publication Date: 10 October 1996 (10.10.96)
(21) International Application Number: PCT/US96/04187	(81) Designated States: AL, AM, AT, AU, AZ, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, EE, ES, FI, GB, GE, HU, IS, JP, KE, KG, KP, KR, KZ, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, TJ, TM, TR, TT, UA, UG, UZ, VN, ARIPO patent (KE, LS, MW, SD, SZ, UG), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).	
(22) International Filing Date: 27 March 1996 (27.03.96)	<p>Published <i>With international search report. Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.</i></p>	
(30) Priority Data: 08/415,559 3 April 1995 (03.04.95) US		
(71) Applicant: ITERATED SYSTEMS, INC. [US/US]; Building 7, Suite 600, 3525 Piedmont Road, Atlanta, GA 30305 (US).		
(72) Inventor: BARNESLEY, Michael, F.; 335 Pennbrooke Trace, Duluth, GA 30136 (US).		
(74) Agent: LOCKMAN, David, M.; Morris, Manning & Martin L.L.P., 1600 Atlanta Financial Center, 3343 Peachtree Road, N.E., Atlanta, GA 30326 (US).		

(54) Title: METHOD AND SYSTEM FOR REPRESENTING A DATA SET WITH A DATA TRANSFORMING FUNCTION AND DATA MASK

(57) Abstract

A system and method are disclosed for representing a data set by selecting a data transformation function and a data masking function. The compressor (10) includes a receiver (12) for receiving data elements of a data set to be represented, a data function generator (14) for generating a data transformation function and attractor, a data mask generator (16) for generating a data masking function, a transform data memory (18) for storing transformed data, and a comparator (20). Data transform function generator (14) selects a data transformation function and uses the data masking function provided by mask generator (16) to generate an attractor. Comparator (20) compares the attractor in memory (18) to the data set in receiver (12) to determine whether the data transformation and masking functions accurately represent the data set in receiver (12). Preferably, the data set transformation and data masking functions are used to generate an attractor that more accurately represents the data set than an attractor generated by the data transformation function alone. The data masking function is a set of exclusionary data elements which are used to terminate data transformations for data elements which generate the attractor. Preferably, the data masking function is defined as a polygon for a two-dimensional space. When the data masking function is incorporated with the data transformation function, the attractor normally produced by the data set transformation function is contained to more accurately represent the data set. The data masking function may be used with any data transformation function including those comprised of complex, projective, affine maps, fractal transform, or fractal transform error function transformations. Deterministic and random iterative methods may be used to generate the constrained attractor that represents the data set to be represented.



**METHOD AND SYSTEM FOR REPRESENTING
A DATA SET WITH A DATA TRANSFORMING FUNCTION
AND DATA MASK**

Field of the Invention

This invention relates to data transforming methods and, more specifically, to data transforming functions used to compress a set of data elements.

5 Background Of The Invention

Bandwidth limitations in various data transmitting media have led to the development of methods for "compressing" data. Compression generally refers to the representation of a set of data by using fewer elements than contained in the set to represent the set of data. If each data element within the set may be regenerated from the compressed representation then the method is commonly known as lossless. If the data representation regenerated from the compressed representation of the data set does not accurately represent each and every element within it, the method is known as lossy. Because the set of compressed data may represent all elements but contain fewer elements than the actual set represented, the transmission of the compressed data usually requires less time and fewer resources than if the original set of data was transmitted. Conversion of the transmitted compressed data set is called decompression and is usually performed by a receiver to obtain the original data set or a close approximation thereto. Thus, the compressed data must not only be capable of accurately representing the data set, but it must also contain sufficient information for the receiver to decompress the data representation.

One known method for compressing a data set uses a data set transformation function to represent a data set. The data set transformation function may be constructed out of projective, affine or complex transformation functions that, when applied to any data set in iterative fashion, converges to the same attractor. As a result, the coefficients of the data set transformation function may be used to represent a data set that is substantially the same as the attractor. Such a method is taught and disclosed in U.S. Patent No. 4,941,193 to Barnsley et al., and is expressly incorporated herein by reference. The method taught and disclosed by that patent yields data compression ratios of greater than 12,000 to 1 in some applications.

Another known image and data compression method uses a fractal transform function to transform data. That method divides an image into range blocks and domains under predetermined conditions. The range blocks are then transformed by a data transformation function. The domain blocks may then be compared to the transformed range blocks to determine which transformed range block best matches each domain. Information about the transformations for each range block and the matching domain blocks are used to represent the original data set. This method is taught and disclosed in U.S. Patent No. 5,065,447 to Barnsley et. al, and is expressly incorporated herein by reference. Alternatively, the transformation data function used to transform the range blocks and the original data set may be used to define an

error function. Minimization of the error function defines coefficients which may be used to represent the original data set.

One problem with data compression methods which use data set transformation functions is the difficulty in efficiently adapting a known attractor or transformed range block so it accurately represents a data set that is somewhat different than the attractor or transformed range block produced by the data transformation function. This problem is especially noteworthy in the conversion of data sets which represent real world physical phenomena. For example, video images or accompanying soundtracks are typically produced at a predetermined rate such as thirty (30) frames a second. As a result, for real-time application there is a time limitation for selecting a data transformation for compressing the data. Previous methods have operated by selecting and modifying a data transformation function so the attractor or transformed range block conforms to an original data set or portion thereof. These methods then collect the data not represented by the attractor or transformed range blocks into a residual set. The residual set of data is transmitted with the coefficients of the data transformation for reconstruction of the original data set at the receiver.

The methods discussed above suffer from, at least, the following limitations. First, the residual data set may be quite large and, as a consequence, the efficiencies normally achieved by using data transformations functions are degraded. Second, the fractal transform method may consume much of the time allocated for compressing and transmitting data with testing the correspondence of a data set to an attractor rather than modifying the data transformation function so the attractor may more efficiently represent the data. Third, the coefficients generated from the minimization of the error function may not accurately depict the data or the real values of the coefficients may require so many digits for accurate representation that the compression achieved is relatively nominal.

What is needed is a way of efficiently adapting a data set transformation function which has a high compression efficiency so the attractor it produces more accurately represents a data set with few or no residual data set elements. What is needed is a way to simplify the representation of the data transformation function so the coefficients of the function may be accurately and efficiently represented.

Summary of the Invention

The above identified problems have been overcome by a novel system and method that combines a data set transformation method with a data masking function. The inventive method includes selecting a data transformation function and a data mask which cooperates with the data transformation function to selectively exclude data elements from the attractor or transformed range block otherwise produced by the data set transformation function. Such a method permits a data transformation function to be efficiently adapted so the attractor or transformed range block more accurately represents the original data set. As a result, a residual data set may not be necessary or the number of elements in such a set may be substantially

reduced. Once an appropriate data transformation function and masking function have been selected, they may be used to more efficiently compress and decompress the data than known methods.

5 The inventive method may be utilized with many data transformation function systems such as those comprised of projective, complex, affine map, fractal transform, or error function fractal transform transformation systems. Such systems may be used in a variety of applications. For example, a data set transformation that produces a graphical shape that resembles a waveform may include extraneous information or non-real world data such as more than one value for a given input. This extraneous information prevents the data transformation
10 function from accurately representing the physical data. By using the system and method of the present invention, the attractor of the data transformation function may be adapted by a masking function to constrain the attractor or transform a range block so it more accurately represents the physical data. Rather than provide data elements which represent discrete values of a soundtrack or pixel elements for a video image, the data set transformation function and
15 masking function may be used to accurately define the data set with a small or no residual data set.

Other applications may include using a data masking function to filter undesired data elements from an original data set. Additionally, a data masking function may be used to enhance or modify aesthetically pleasing designs created by a computer driven generator to
20 expand the permutations of designs possible with such a generator.

To decode data compressed with data transformation and masking functions, a receiver simply uses the coefficients of the transformation function and the definition of the data masking function to reproduce the attractor or transform a data set using a fractal transform. The inventive methods for generating an attractor using a data masking function include a
25 random iteration method, a deterministic method, and a fractal transform method. The attractor may then be used to reproduce the original data set such as a video frame or soundtrack waveform. Thus, physical data such as image and sound data may be more efficiently communicated between locations. By combining a masking function that excludes data with one that transforms data within a data space, data sets may be efficiently compressed. That is,
30 the number and size of the data transformation function coefficients and the size of a residual data set may be substantially reduced.

These and other features and advantages of the present invention may be more clearly understood and appreciated from a review of the following detailed description and by
35 reference to the appended drawings and claims.

Brief Description of the Drawings

The present invention may take form in various components and arrangement of components and in various steps and arrangement of steps. The drawings are only for purposes

of illustrating embodiments of the invention and are not to be construed as limiting the invention.

Fig. 1 is a block diagram of a data compressor which incorporates the principles of the present invention;

5 Fig. 2 is an example of the results of a data transformation function acting on a data set;

Fig. 3 is block diagram of a data decompressor which incorporates the principles of the present invention;

Fig. 4 is a block diagram of a data compressor which incorporates the principles of the present invention for a fractal transform method;

10 Fig. 5 is block diagram of a data decompressor which incorporates the principles of the present invention for a fractal transform method;

Fig. 6 illustrates an object accurately represented an attractor produced by the data transforming function shown in Fig. 7;

15 Fig. 7 illustrates the affine maps which may be used to generate an attractor that represents the object shown in Fig. 6;

Fig. 8 illustrates an object having structure not accurately represented by the attractor generated by the affine maps shown in Fig. 7;

Fig. 9 illustrates a data masking function that may be incorporated with the affine maps of Fig. 7 to generate an attractor that more accurately represents the object depicted in Fig. 8;

20 Fig. 10 is a logic diagram of a deterministic process used to generate an attractor from a data transformation and data masking functions of the present invention;

Fig. 11 is a logic diagram of a random process used to generate an attractor from a data transformation and data masking functions of the present invention; and

25 Fig. 12A is a logic diagram of a fractal transform process used to encode a data set using data transformation and data masking functions to transform range blocks within the fractal transform process; and

Fig. 12B is a logic diagram of a modification to the process shown in Fig. 12A.

Detailed Description of the Invention

30 A data compressor incorporating the present invention is shown in Fig. 1. The compressor 10 includes a receiver 12 for receiving data elements of a data set to be represented, a data function generator 14 for generating a data transformation function and attractor, a data mask generator 16 for generating a data masking function, a transform data memory 18 for storing transformed data, and a comparator 20. Data transform function generator 14 selects a
35 data transformation function and uses the data masking function provided by mask generator 16 to generate an attractor. Comparator 20 compares the attractor in memory 18 to the data set in receiver 12 to determine whether the data transformation and masking functions accurately represent the data set in receiver 12.

A data set transformation function used by data function generator 14 may be singular or a set of transformation functions. Each transformation function has a finite set of coefficients. Examples of such data set transformation functions include complex, affine, or projective transformations. Any of the data set transformation functions generated may be either lossy or lossless data set transformation functions. Function generator 14 may also select an iterated function system to generate an attractor by using the method disclosed in U.S. Patent No. 4,941,193. Selection of the coefficients for any data set transformation function may be under the control of either an operator or a predetermined selection method.

An example of a data set transformation acting on a data set is shown in Fig 2. There an original data set, B, includes the pixel elements which graphically represent the image shown in the figure. A data set transformation function, S, may be comprised of a plurality of data element transformations, W_i for $i = 1, 2, 3 \dots N$, which may act on the data elements of the original data set to generate a new data set. The image in Fig. 2 is transformed by three transformations comprising a data transformation function to produce the image 20.

The operation of a transformation function on set B is symbolically represented as $S(B) = \bigcup_{i=1}^3 W_i(B)$. That is, a transformation of a data set B is equal to the union of the operation of the three data transformations on the original set B. If the data set transformation function S is iteratively applied to the previous S(B) then the resulting transformations may converge to an attractor. Certain attractors may be generated by the iterative operation of a transformation function regardless of the data set on which the function first acts. In such a case, the coefficients of the transformation function, S, may be used to represent the attractor. Because the elements necessary to define the coefficients of the transformations comprising S are usually less in number than the elements of the original data set, S is a data transformation that compresses the original data set. Another possible data function is a data transformation function with a condensation set. Such systems are well-known.

Data function generator 14 uses the data masking function received from data mask generator 16 to transform data elements through one or a combination of the transformation processes discussed in more detail below. Additionally, data function generator 14 may include memory for receiving data elements from receiver 12 which may be analyzed to select a data transformation function. Data function generator 14 also includes a processor for receiving error distance measurements or minimized error function coefficients from comparator 20 which may be used in accordance with known methods to modify the coefficients of a data transformation function. Alternatively, data function generator 14 may incorporate the transform memory 18.

Mask generator 16 of Fig. 1 selects a data masking function which cooperates with a data set transformation function in the processes discussed below to constrain the attractor normally produced by a data transformation function. The data masking function defines a set of exclusionary data elements. These data elements are used to control the application of the

data transformation function to a set of data elements so the attractor otherwise generated by the transformation function altered. Thus, the attractor may be altered to more accurately represent the original data set by modifying either the data transformation function or the data masking function or both. The data masking function may be defined by an operator using a mouse as discussed in more detail below or selected from a set of masking functions based on an analysis of the data set to be represented. The data contained in receiver 12 may be displayed on a screen device or the like (not shown) to assist an operator in interactively selecting data transformation and data masking functions.

Likewise, data mask generator 16 includes a processor for receiving error distance measurements or minimized error function coefficients from comparator 20 which may be used to alter the data masking function. For example, where an attractor contains extraneous data, the exclusionary set of data elements defining the data mask may be expanded to include more elements to further constrain the attractor. Likewise, if the attractor lacks data elements present in the data set to be represented, then some of the data elements comprising the data mask may be removed. The addition or deletion of data elements in the data mask may be done interactively, for example, by displaying the mask and attractor over the data set to be represented and allowing the operator to redefine the data mask by redefining a set of elements graphically. Alternatively, the processor may alter the function description of the mask to change the data mask definition. For example, if the data mask is defined as a polygon, the mathematical function describing the polygon may be changed to reduce or increase the perimeter or shape of the polygon.

Transform data memory 18 and comparator 20 are used to evaluate the data transformation and data masking functions selected by function generator 14 and data mask generator 16, respectively. To perform the evaluation, the data space of transform data memory 18 is transformed to an attractor generated by the selected data transformation and data masking functions which preferably operate according to one of the processes discussed in detail below. Comparator 20 compares the attractor in memory 18 to the original data in memory 12 to calculate an error measurement. Preferably, the error measurement may be a Hausdorff distance between the data set in receiver 12 and the attractor in memory 18. Such a measurement is well known. Other possible error measurements which may be used are Hamming and root mean square measurements.

The error measurement is preferably provided to data transformation function generator 14 for modifying the data transformation function or to the data masking generator 16 for modifying the data masking function. These modifications are preferably optimized by generating an error function which may be minimized by using a neural network technique, an annealing method, or genetic algorithm, all of which are well-known. Alternatively, a display may be generated which overlays the attractor over the original data set so an operator may

interactively modify the data transformation function or data masking function to improve the correspondence between the two as discussed above.

If the error measurement is less than a predetermined threshold, comparator 20 generates a signal indicative that the selected data transformation and masking functions adequately represent the original data set. Following receipt of that signal, data function generator 14 and masking function generator 16 transfer the coefficients which define the data transformation and the coefficients or data elements which define the data masking function to transmitter 22 for transmission to a remote site for decompression.

A decompressor 30 for the inventive system 10 of Fig. 1 is shown in Fig. 3. The decompressor 30 includes a receiver 32 for receiving the data defining the data set transformation and data masking functions, a data transformer 34 for using the data set transformation and masking functions to generate the attractor which represents the original data set, and a memory 36 for storing the generated attractor. The data transformer preferably uses one of the processes discussed in more detail below to regenerate the attractor that represents the original data set. The data stored in memory 36 may be used to drive a video display, sound reproducing components, or other data applications.

A block diagram for an embodiment of the present invention which may be used with a fractal transform method is shown in Fig. 4. That system 40 includes a receiver 42, a data processor 44, transform memory 46, and transmitter 48. The receiver may be any type of device which receives data elements for a data set to be represented. Transform memory 46 includes two buffers. Buffer 50 is used for storage of domain blocks and buffer 52 is used for storage of range and transformed range blocks. Transmitter 48 is used to transfer the transformed data which represents the data set to another device.

Data processor 44 includes control logic 54, block comparator 56, mapping processor 58, and data masking function generator 60. Control logic 54 executes programs which initiate block comparisons and mapping functions in mapping processor 58. Specifically, after a data set has been received by receiver 42, control logic 54 divides the data set into domain and range blocks which are stored in buffers 50, 52, respectively.

Additional range blocks are generated by mapping processor 58 using data transformations functions. Preferably, the data transformations used by mapping processor 58 include pixel number, pixel value, or pixel arrangement transformations. Pixel number transformations include contractive mapping functions, pixel group averaging functions and the like. Pixel value transformations include attenuation of pixel value functions, inverting pixel value functions, or the like. Arrangement transformations include rotation of pixel group functions or the like. Mapping processor 58 also uses data masking functions provided by data masking generator 60 to transform the range blocks. These transformed range blocks are stored in buffer 52 for comparison to the domains stored in buffer 50. By using the data masking function of the present invention with a data transformation function, the mapping processor 58

may generate transformed range blocks which more closely match the domains. As a result, the fractal transforms generated by system 40 more accurately represent the data set in receiver 42.

5 A decompressor 70 for system 40 of Fig. 4 is shown in Fig. 5. The decompressor 70 includes a receiver 72 for receiving the data defining the data set transformation and data
5 masking functions, a data transformer 74 for using the data set transformation and masking functions to generate the attractor which represents the original data set, a memory 76 for storing transformed data which converges to the attractor, and a transmitter 78 for delivering the data elements of the generated attractor to another device. The data elements of the attractor
10 may be used to drive a video display, sound reproducing components, or other data applications.

Data transformer 74 includes control logic 80, a count buffer 82, and a pattern generator 84. Control logic 80 executes a program to retrieve the identifiers for the data transformation and data masking functions along with the range block identifiers from receiver 72. A buffer 86 is initialized with predetermined set of data elements provided by pattern generator 84. Control
15 logic 80 transforms the predetermined data set in buffer 86 by using the data transformation and data masking functions identified by the identifiers retrieved from receiver 72 and stores the transformed data elements in buffer 88. The transformation process is repeated using buffer 88 as the source and storing the transformed elements in buffer 86. This process continues for the number of times stored in count buffer 82. At that time the attractor in the buffer containing the
20 most recently transformed data elements represents the original data set.

Preferably, the systems described above which use the processes described below are implemented in Visual Basic which runs under a Windows operating system, available from Microsoft of Redmond, Washington, in enhanced mode. A computer system which supports such an implementation requires an Intel 80386 processor or equivalent with at least 4
25 megabytes of random access memory (RAM) and a disk drive having at least 20 megabytes of space available. Any monitor compatible with such a system may be used.

An attractor which resembles a real world object, a black spleenwort fern, is shown in Fig. 6. This attractor may be generated by a known data transformation. As can be observed from Fig. 6, the attractor appears to have an infinite number of fronds as it progresses to its
30 lowest end and each frondlet of each frond appears to have an infinite number of substructures at the end of each one. Attractors having this type of structure arise from iterative transformation methods because successive iterations of data transformations result in data values which tend to approach a finite value even if the iterations continued infinitely. For that reason, the additional change from one iteration to the next may be measured and when the
35 overall difference falls below a predetermined threshold, the iterations may be terminated. For example, the rate of change in a Hausdorff distance from one iteration to the next in the generation of an attractor may be used to determine when a sufficient number of iterations have been performed.

The affine maps, A, B, C, and D, of a data set transformation function which may be iteratively applied to a data set to generate the attractor in Fig. 6 are shown in Fig. 7. A number of methods may be used to generate the attractor such as the random iterative and deterministic methods discussed more fully below. The method and system for selecting these affine maps and for generating an attractor without the use of a data masking function are disclosed in U.S. Patent No. 4,941,193.

The object depicted in Fig. 8 is not accurately represented by the data set transformation function comprised of the affine maps shown in Fig. 7. As one can observe from the figure, this fern has a finite number of fronds and each frond has a finite number of frondlets. If the coefficients of the data transformations represented by the maps of Fig. 7 are modified, an attractor having a structure which asymptotically approaches a limit at the boundaries of the object still emerges. Nor could the truncated boundaries be generated by simply limiting the number of iterations to produce the attractor since the iterations do not necessarily produce the attractor from the interior of the object to the boundaries. While the fern of Fig. 8 contains less information than the fern of Fig. 6, there currently is no known way of limiting the attractor and still efficiently represent the transformation function that generates the attractor.

The system and method of the present invention constrains the attractor in Fig. 6 by use of a novel data masking function. The masking function is a set of data elements which are used in the data transformation method to alter the process so as to exclude data elements from the attractor during data element transformation. The masking function may include a geometric shape or simply a collection of arbitrary data elements in the transform memory. The purpose of the masking function is to define an exclusionary data set in the data transform space for the iterative process.

An example of such a masking function or data mask is shown by the shaded region 92 in Fig. 9. Such a mask may be defined by a mathematical equation or it may be defined by an operator clicking on points in a display to form the polygon that defines the data masking region. Alternatively, a user could define the mask by identifying the x, y coordinates or the like that define the vertices of a polygon or simply identifying the data set elements which form a data masking function. Preferably, a system that permits an operator to define the masking function by using a mouse includes a display so the operator may view the data set to be reproduced and the selected mask. Thus, the user would be able to manipulate the mouse more easily at the vertices which appear to be related to the boundary of the data set to be represented. Alternatively, the data mask generator 16 may randomly select a data masking function or select a data masking function that complements a data set transformation function selected by data function generator 14. For example, a data masking function that follows the boundaries of a data transformation function or which fits between data transformations comprising a data transformation function may be selected by data mask generator 16.

The mathematics which support the use of a data mask may be stated as follows. Let K denote an n -dimensional space which may be represented as R^n where $n = 1, 2, 3, \dots$. A masked data transformation function system of order N consists of a space K together with a collection of N continuous, contractive transformation functions $\{W_i : i = 1, 2, \dots, N\}$ from K to K together with the definition of an exclusionary set which may be represented by a line when $N=1$, a polygon when $N=2$, or a polytope when $N>2$. Alternatively, the exclusionary set may be comprised of data space elements in N space which can be as simple or as complex a subset of K as desired.

For example: if $n = 2$ then $K = R^2$ (Euclidean space) and the W_i are often of the form:

$$W \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}$$

The symbols a, b, c, d, e and f are real constants or coefficients which specify the function. These transformation functions are merely exemplary as the invention is envisioned as being applicable to other transformations such as complex or non-linear transformations within the space K . A continuous function $W: K \rightarrow K$ is said to be mathematically contractive if it always decreases the distance between points. Let the distance between two points x and y in the space K be denoted by $|x-y|$ then W is contractive with contractivity factors such that $0 \leq \epsilon < 1$ if $|W(x)-W(y)| \leq \epsilon |x-y|$, for all pairs of points (x,y) in K . In the example transformation noted above, the six real coefficients are chosen to ensure that all the W_i are contractive within R^2 .

The examples above assumed that the space K was made up of physical points. For example: R^1 is a real line and R^2 is an Euclidean plane. However, the space K may be any compact metric space which is a term well known

to those skilled in the art. For example the space K could be the space in which each point is a continuous function such as a density or color intensity for a grayscale or color image pixel .

Let an iterative system be applied to a set of points A in K to transform it to a new set of points A':

$$A' = \bigcup_{i=1}^N W_i(A)$$

If $A = A'$ then an invariant set of points has been found that form an attractor under this iterative system. U.S. Patent No. 4,491,193 shows that an unique invariant set of points exists under suitable conditions of contractive factors for the W_i . This invariant set of points is referred to as the attractor for the data set transformation function comprised of the data transformations. An automatic method for selecting a set of data transformations using a fractal transform method is also shown in U.S. Patent No. 5,065,477 and this method may also be used with the present invention.

A simple data mask may be a collection of (x,y) coordinate pairs which define a simple, closed polygon. Let $M \subset K$ be a data mask so defined. Then the complement of the mask is $M^c = K \setminus M$, where the symbol "\setminus" is the set subtraction operator. That is, M^c is the set of data elements in space K which are not within the mask polygon. Applying a data transformation function with a data mask yields:

$$A' = \bigcup_{i=1}^N W_i(M^c \cap A)$$

If $A' = A$ then an invariant set of points has been found that form an attractor under the iterative data transformation and data mask system.

The data transformation and data masking functions may be used in a deterministic process to generate an attractor for representing an original data set by the method shown in Fig. 10. The method begins by setting an iteration count to a predetermined maximum value (Step 100) and initializing a data buffer with a random data set (Step 102). The masking data function is used to identify data elements in the data buffer which are not transformed. The process continues by applying every data transformation of a transformation function to data elements of the random data set in the data buffer which are not in the data masking function (Steps 104-116). The resulting transformed data element for each applied data transformation is stored in a data transform buffer (Step 112). After each data transformation has been applied to each data element of the random data set, the initial iteration is complete. The iteration count is then reduced by one. (Step 118). If the count is non-zero, the data buffer is cleared and the data transformations are applied to the data elements in the data transform buffer which are stored in the data buffer. (Steps 120-122). This may be done by exchanging the content of the buffers or a flag identifying the buffers or any known method for altering the source and destination buffers for a process. The steps of storing transformations of the data elements of one data buffer in the other data buffer is repeated until the iteration count is zero. (Step 112).

At that time, the attractor that represents the original data set is stored in the data buffer containing the latest transformed data elements.

Alternatively, the process may measure an error difference between the data and transform buffer after all the elements of one buffer have been transformed. If the error difference exceeds a predetermined threshold then the data elements of the buffer in which the transformed elements were stored are then transformed and stored in the other buffer. If the error difference is less than or equal to the threshold, the transformation of the data elements in the buffers is halted and the last stored buffer continues the attractor representing the original data set. Otherwise, the buffer storing the most recently transformed elements are transformed using the data transformation function and data masking function and the resulting transformed elements stored in the other buffer. The error distance may be measured by one of a Hausdorff, Hamming, or least squares distance or equivalent method.

A random iteration process which may be used to generate the attractor from the coefficients for the data transformation and data masking functions is shown in Fig. 11. That method begins by initializing an iteration count to a predetermined value (Step 150) and a random data element in a data buffer (Step 152). The process continues by randomly selecting and applying one of the transformations of the data transformation function. (Step 154-156). The transformations comprising the transformation function may be assigned probabilities and these probabilities may be used to randomly select the transformations. The transformed data element is then stored in the data buffer (Step 158) and tested to see if it lies within the data masking set. (Step 160). If the data element is within the masking function, the process selects the last transformed data element stored in the data buffer (Step 162) before it determines whether to continue the random selection and application of data transformations to data elements. (Step 164-166). Thus, the data masking function is used to halt the propagation of transformed data elements from the members of the exclusionary data set. After a predetermined number of iterations have been performed (Steps 164-166), the process terminates and the attractor representing the original data set is contained in the data buffer.

The data masking function may also be used with the fractal transform method disclosed in U.S. Patent No. 5,065,477. Incorporation of the data masking function in this method to encode data is shown in Fig. 12A. First, a data buffer is initialized with a data set (Step 200) which is divided into a plurality of domains which conform to certain requirements. (Step 202). The image is also used to generate a plurality of range blocks which must be larger than the domains. (Step 204). Additional range blocks are generated by using data transformation functions with a data masking function. For the transformations performed with the data masking function, a data transformation function is applied to the data elements of a range block which are not in the masking function and the transformed elements are stored in the transformed range block. (Steps 206-212). The process continues after the range blocks have been transformed by determining for each domain the range block which best matches the

domain. (Steps 216-226). The data transformation function and data masking function for that range block, if used to transform the range block, are used to represent the domain block.

- After a data transformation function and masking function have been selected for each domain, the coefficients are then transmitted to a receiver. The receiver performs the process discussed above with respect to system 70 to decompress each domain and regenerate an attractor representing the original data set. The match is determined by mapping a range block to a domain and computing an error measurement as discussed previously. The smallest error measurement indicates the best range block match. The use of the data masking function with the transformation function provides transformed range blocks which better represent the domains of the original data set.

- The data masking function may be incorporated in the process shown in Fig. 12 in another way to more accurately represent data. The steps shown in Fig. 12B may be incorporated between steps 216 and 220. Such a process may or may not use a data masking function to transform range blocks. The steps shown in Fig. 12B use a masking function by identifying range blocks which are used in the comparison of a domain to a range block. This masking function may be differently defined for each domain.

The mapping of a range block to a domain in a data space where $N=3$ may be represented by the equation:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \rightarrow \begin{bmatrix} w & 0 \\ \vdots & \vdots \\ a & b \\ \vdots & \vdots \\ c & \vdots \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ d \end{bmatrix}$$

- where W is a two dimensional range block transformation and t_x , t_y and d are translations in the x , y and z axes, respectively. A function representing an error distance between a domain and range block may be expressed in a least squares measurement, $G(a,b,c,d)$, although other error measurements may be used. Thus,

$$G(a, b, c, d) = \int_{\text{domain}} [f(w(x, y)) - ax - by - cf(x, y) - d]^2 dx dy$$

- where $z = f(x,y)$ is the gray scale intensity. This error measurement may be minimized by taking the partial derivative of the measurement with respect to each coefficient a, b, c , and d , setting each partial derivative equal to zero and solving the linear system of the four equations for the four unknowns, namely, the coefficients a, b, c , and d . These values may then be used to represent the data set of the range block.

- The masking function of the present invention may be used to improve the match of the domain to range blocks by defining a domain so it more accurately conforms to a data subset having data elements which are related to one another. For example, in consecutive video frames a domain of an N^{th} frame may be compared to a range block of a $N-1^{\text{th}}$ frame. If domains conform well to range blocks within prior video frames, then the data masking function may be used to reduce the number of range blocks tested for an error measurement. If

the difference between consecutive frames exceeds a predetermined threshold, then the set of difference values may be represented by a data transformation function and the time for calculating the coefficients may be reduced. Similarly, the masking function may be used in a data space application to exclude address spaces for data not relevant to an application. In this manner, a data element may be found in an address space without a tree search or other data element locating search.

Each of the methods set forth above may be used to produce resolution independent images. That is, regardless of the detail scale chosen for the image, no information content is lost. This property of iterated system and fractal transform systems are well know. However, the incorporation of the data masking function of the present invention more accurately provides detail in such images.

In operation, receiver 12 of system 10 in Fig. 1 receives a set of data elements from a source for compression. The data set may be a frame of video data, a PET image, a soundtrack segment that accompanies an image, or a schema for a database. A data transformation function is selected by data transform generator 14 and a data masking function is selected by data mask generator 16. The data transformation function and data masking function may be selected by interaction of an operator with system 10 or by the selection methods discussed above. Using the selected data transformation and data masking function, data transform generator 14 generates an attractor for representing the original data set received by receiver 12. The attractor may be generated, for example, by a random or deterministic method for an iterated function system, or it may be generated by a fractal transform method, or any other known method for generating attractors.

After the attractor is generated, the attractor is evaluated to determine whether it accurately represents the original data set. One way to evaluate the attractor is to compute the Hausdorff distance between the original data set and the attractor. If this measurement is below a predetermined threshold the coefficients for the selected data transformation and data masking functions may be transmitted to represent the original data set. If the measurement is greater than the predetermined threshold, the coefficients of the data transformation or the data masking function or both may be modified. After modification, the attractor is regenerated and the comparison test is repeated to determine whether the threshold has been met. The results of that test may be used to determine whether the modifications enhanced or degraded the accuracy of the attractor and that information may be used to select the next modification, if one is needed.

After appropriate data transformation and data masking functions are selected, the coefficients of the functions may be transmitted or, if a fractal transform method is used, the coefficients of the data transformations and data masking functions for the domains and the range data information may be transmitted to represent the data. The data compressor located at the receiver then performs the attractor generation process performed by data transform

generator 14 to generate the attractor that represents the original data set. The data of the attractor may then be used to drive a video imager, a sound system, a database system or the like.

5 Those skilled in the art will readily appreciate the various modifications and variations which may be made to the system and methods disclosed in this patent without departing from the spirit or scope of the present invention. For example, the system and method of the present invention may be adapted to incorporate data masking functions with complex, non-linear data transformations. As a result, the system and method should not be limited to affine transformations or to iterated function systems or to fractal transform methods. The present
10 invention covers such modifications and variations which are within the scope of the appended claims and their equivalent.

What is claimed is:

The claims defining the invention are as follows:

1. A method for generating a compressed representation of a data set, said method comprising the steps of:
 - 5 receiving said data set to be represented,
 - selecting a data set transformation function that converges, upon iterative application, to an attractor that approximates said data set,
 - selecting a data masking function defining a set of exclusionary data elements for constraining said attractor normally produced by said data set transformation
 - 10 function whereby the constrained attractor generated by the selected data set transformation and data masking functions more accurately represents said data set, and
 - generating coefficients which define the data set transformation function and coefficients or data elements which define the data masking function as said compressed representation of said data set.
- 15 2. The method of claim 1 said selecting data set transformation function step further includes the step of:
 - selecting a plurality of data transformations to represent said data set.
- 20 3. The method of claim 1 said selecting data set transformation function step further comprising the step of:
 - selecting one of a complex, affine, and projective transformation function.
- 25 4. The method of claim 1 said selecting data masking function step further includes the step of:
 - selecting a set of exclusionary data elements.
- 30 5. The method of claim 4 said selecting data exclusionary set including the step of:
 - selecting a polygon as said data masking function.
- 35 6. The method of claim 2 said selecting transformations step further includes the step of:
 - selecting coefficients for each selected transformation.
7. The method of claim 1 further comprising the step of:
 - evaluating said constrained attractor to determine an error between said constrained attractor and said data set to be represented; and

modifying one of said selected data set transformation function and said selected data masking function in response to said determined error being greater than a predetermined threshold.

5 8. The method of claim 7, said modifying data set transformation function step further includes the step of:
 modifying at least one coefficient of said selected transformation function.

 9. The method of claim 8, said modifying selected transformation
10 function and said data masking function further includes the step of:
 modifying data elements of an exclusionary data set comprising said data masking function.

 10. The method of claim 8 further comprising the step of:
15 determining whether said modified data transformation function and said selected data masking function generate an attractor that more accurately represents said data set.

 11. The method of claim 10 further comprising the step of:
20 transmitting said coefficients of said modified data transformation function and said data masking function so that a receiver may generate said attractor that represents said data set.

 12. The method of claim 1 said selecting data set transformation function
25 step further including:
 selecting a fractal transform method to transform said data set.

 13. The method of claim 12 further including the step of:
 selecting a data masking function to constrain an attractor generated by
30 transforming a range block.

 14. The method of claim 1, further including the step of:
 generating an error function which defines an accuracy error between said data set and said data transformation and data masking function; and
35 minimizing said generated error function to reduce a set of coefficients to represent said data set.

15. A method for generating a transformed data set which represents a data set comprising the steps of:

initializing a data buffer with a random image;

5 applying a data transformation function transforming data contained in the data buffer to each data element of said random image and storing said transformed data element in a transform buffer unless said data element of said random image is within a data masking function defining a set of exclusionary data elements;

measuring an error difference between said data elements of data buffer and said transform buffer;

10 exchanging said data buffer and said transform buffer; and

repeating said applying, and measuring, and said exchanging steps until said error measurement does not exceed a predetermined threshold so that the last buffer to which data elements were stored contains an attractor generated by at least said data transformation and data masking functions and representing said data set.

15

16. The method of claim 15 wherein said error difference measuring step measures said error difference by computing one of a Hausdorff distance, a root mean square distance, and a Hamming distance.

20

17. The method of claim 15 further comprising the steps of:

measuring an error difference between said attractor and said data set being represented; and

25 modifying one of said data transformation function and said data masking function whereby said attractor generated by said modified data masking function and data transformation more accurately represent said data set.

30

18. The method of claim 17 wherein said error difference measuring step measures said error difference by computing one of a Hausdorff distance, a root means square distance, and a Hamming distance.

35

19. A method for generating a transformed data set which represents a data set comprising the steps of:

initializing a data buffer with a random image;

35 applying a data transformation function transforming data contained in the data buffer to each data element of said random image and storing said transformed data element in a transform buffer unless said data element of said random image is within a data masking function defining a set of exclusionary data elements;

decrementing an iteration count;

exchanging said data buffer and said transform buffer; and
repeating said applying and said decrementing steps until said iteration count is
zero so that said last buffer to which data elements were stored contains an attractor
generated by at least said data transformation and data masking functions and
5 representing said data set.

20. The method of claim 18 further comprising the steps of:
measuring an error difference between said attractor and said data set being
represented; and
10 modifying one of said data transformation function and said data masking
function whereby said attractor generated by said modified data masking function and
data transformation more accurately represent said data set.

21. The method of claim 20 wherein said error difference measuring step
15 measures said error difference by computing one of a Hausdorff distance, a root mean
square distance, and a Hamming distance.

22. A method of generating a transformed data set which represents a data
set comprising the steps of:
20 initializing a data element in a data buffer;
applying a data transformation function to transform said data element within
said data buffer;
determining whether said transformed data element is within a data masking
function defining a set of exclusionary data elements;
25 applying said data transformation function to one of said transformed element
and a last transformed element in response to said determination of said transformed
data element being within said data masking function;
decrementing an iteration count; and
continuing said determining and applying steps until said iteration count is zero
30 whereby said transformed data elements in said data buffer form an attractor that is
generated by at least said data transformation and data masking functions and that
represents said data set.

23. The method of claim 22 further comprising the step of:
35 selecting a transformation from a set of transformations that comprise said data
transformation function.

24. The method of claim 23 wherein said transformation is selected in accordance with a probability corresponding to said transformations comprising said transformation function.

5 25. The method of claim 23 wherein said set of transformations is a plurality of affine maps.

26. The method of claim 22 further comprising the step of:
10 defining said data masking function as a set of exclusionary data elements.

27. The method of claim 26 said defining step further includes the step of:
selecting vertices which define a polygon.

28. The method of claim 27 said selecting vertices step further includes
15 the step of:
selecting said vertices being done interactively.

29. The method of claim 27 wherein said selecting vertices step further
20 includes the step of:
identifying coordinates for said vertices.

30. The method of claim 27 wherein said selecting vertices step further
25 includes the step of:
selecting a mathematical equation that defines said polygon.

31. The method of claim 23 wherein said set of transformations include
one of a set of projective, complex, and affine transformations.

32. The method of claim 22 further comprising the steps of:
30 measuring an error difference between said attractor and said data set being
represented; and
modifying one of said data transformation function and said data masking
function whereby said attractor generated by said modified data masking function and
data transformation more accurately represent said data set.

35 33. The method of claim 32 further comprising the step of:
modifying coefficients of an affine map in response to a determination that said
attractor does not conform to said set of data.

34. The method of claim 32 further comprising the step of:
modifying vertices of a polygon defining said data masking function in
response to a determination that said attractor does not conform to said data set.

5

35. A system for generating an attractor from a data transformation
function and a data masking function that represent a data set comprising:
a data function generator for selecting a data set transformation function that
converges, upon iterative application, to an attractor that approximates said data set,
10 a data mask generator for selecting a data masking function defining a set of
exclusionary data elements for constraining said attractor normally produced by said
data set transformation function,
said data function generator further generating the constrained attractor using
the selected data set transformation and data masking functions, and
15 a transmitter for transmitting said data set transformation function and said
data masking function to a remote site for regeneration of said constrained attractor.

36. The system of claim 35 further comprising:
a comparator for comparing said attractor generated by said data function
20 generator to a data set to be represented by said attractor, said comparator measuring an
error distance between said data set and said attractor.

37. The system of claim 36 said data function generator further
comprising:
25 a coefficient modifier for modifying coefficients of said data transformation
function in correspondence with said error distance.

38. The system of claim 36 said data mask generator further comprising:
a masking modifier for modifying said data masking function in
30 correspondence with said error distance.

39. The system of claim 36 said comparator further comprising:
an error function generator for generating an error function and minimizing
said error function so that coefficients for said minimized error function may be used to
35 represent said data set.

40. The system of claim 35 wherein said data function generator generates
said attractor with a deterministic process.

41. The system of claim 35 wherein said data function generator generates said attractor with a random iteration process.

5 42. The system of claim 35 wherein said data function generator generates said attractor with a fractal process.

43. The system of claim 35 wherein said data function generator generates said attractor with an error function fractal process.

10

44. The system of claim 42 further comprising:
means for transforming range blocks with another data transformation and another data masking function whereby said transformed range blocks more accurately conform to domains generated from said data set.

15

45. The system of claim 42 further comprising:
means for using said data masking function to prevent comparison of a domain to a range block.

20

46. A system for generating a compressed representation of a data set, comprising:

means for receiving said data set to be represented,
means for selecting a data set transformation function that converges, upon iterative application, to an attractor that approximates said data set,
25 means for selecting a data masking function defining a set of exclusionary data elements for constraining said attractor normally produced by said data set transformation function whereby the constrained attractor generated by the selected data set transformation and data masking functions more accurately represents said data set, and

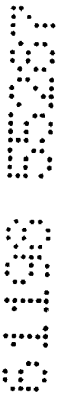
30

means for generating coefficients which define the data set transformation function and for generating coefficients or data elements which define the data masking function as said compressed representation of said data set.

35

47. The system of claim 46 said means for selecting said data set transformation function further comprising:

means for selecting a plurality of data transformations to represent said data set.



48. The system of claim 46 wherein said selected data set transformation is one of a complex, affine, and projective transformation function.

49. The system of claim 46 wherein said data masking function is a set of
5 exclusionary data elements.

50. The system of claim 49 said data exclusionary set being a polygon.

51. The system of claim 46 further comprising:
10 means for selecting coefficients for each selected transformation.

52. The system of claim 46 further comprising:
means for evaluating said constrained attractor to determine an error between
said constrained attractor and said data set to be represented; and
15 means for modifying one of said selected data set transformation function and
said selected data masking function in response to said determined error being greater
than a predetermined threshold.

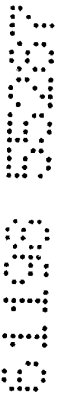
53. The system of claim 52, said means for modifying further comprising:
20 means for modifying at least one coefficient of said selected transformation
function.

54. The system of claim 52, said modifying means further comprising:
means for modifying data elements of an exclusionary data set comprising said
25 data masking function.

55. The system of claim 52 further comprising:
means for determining whether said modified data transformation function and
said selected data masking function generate an attractor that more accurately represents
30 said data set.

56. The system of claim 46 further comprising:
means for transmitting said coefficients of said modified data transformation
function and said data masking function so that a receiver may generate said attractor
35 that represents said data set.

57. The system of claim 46, further comprising:



means for generating an error function which defines an accuracy error between said data set and said data transformation and data masking function; and means for minimizing said generated error function to reduce a set of coefficients to represent said data set.

5

58. A system for generating a transformed data set which represents a data set comprising:

means for initializing a data buffer with a random image;

10 means for applying a data transformation function transforming data contained in the data buffer to each data element of said random image and storing said transformed data element in a transform buffer unless said data element of said random image is within a data masking function defining a set of exclusionary data elements;

means for measuring an error difference between said buffer and said transform buffer;

15 means for applying a data transformation function transforming data contained in the transform buffer to each data element of said transform buffer and storing said transformed data element in said data buffer unless said data element of said transform buffer is within a data masking function defining a set of exclusionary data elements in response to said error difference exceeding a predetermined threshold;

20 means for repeating said applying and said measuring steps until said error measurement does not exceed said predetermined threshold so that the last buffer to which data elements were stored contains an attractor generated by at least said data transformation and data masking functions and representing said data set.

25 59. The system of claim 58 wherein said error difference measuring means measures said error difference by computing one of a Hausdorff distance, a root mean square distance, and a Hamming distance.

60. The method of claim 58 further comprising:

30 means for measuring an error difference between said attractor and said data set being represented; and

means for modifying one of said data transformation function and said data masking function whereby said attractor generated by said modified data masking function and data transformation more accurately represent said data set.

35

61. A system for generating a transformed data set which represents a data set comprising:

means for initializing a data buffer with a random image;

means for applying a data transformation function transforming data contained in the data buffer to each data element of said random image and storing said transformed data element in a transform buffer unless said data element of said random image is within a data masking function defining a set of exclusionary data elements;

5 means for decrementing an iteration count;

means for applying a data transformation function transforming data contained in the transform buffer to each data element of said transform buffer and storing said transformed data element in said data buffer unless said data element of said transform buffer is within a data masking function defining a set of exclusionary data elements in response to said decremented iteration count being non-zero; and

10 means for repeating said applying and said decrementing steps until said iteration count is zero so that said last buffer to which data elements were stored contains an attractor generated by at least said data transformation and data masking functions and representing said data set.

15

62. The system of claim 61 further comprising the steps of:

means for measuring an error difference between said attractor and said data set being represented; and

20 means for modifying one of said data transformation function and said data masking function whereby said attractor generated by said modified data masking function and data transformation more accurately represent said data.

63. A system of generating a transformed data set which represents a data set comprising:

25 means for initializing a data element in a data buffer;

means for applying a data transformation function to transform said data element within said data buffer;

means for determining whether said transformed data element is within a data masking function defining a set of exclusionary data elements;

30 means for applying said data transformation function to one of said transformed element and a last transformed element in response to said determination of said transformed data element being within said data masking function;

means for decrementing an iteration count; and

35 means for continuing said determining and applying steps until said iteration count is zero whereby said transformed data elements in said data buffer form an attractor that is generated by at least said data transformation and data masking functions and that represents said data set.

64. The system of claim 63 said set of transformations being one of a set of projective, complex, and affine transformations.

65. The system of claim 63 further comprising the steps of:
6 means for measuring an error difference between said attractor and said data set being represented; and
means for modifying one of said data transformation function and said data masking function whereby said attractor generated by said modified data masking function and data transformation more accurately represent said data set.

10 66. The system of claim 65 further comprising:
means for modifying coefficients of said data transformation in response to a determination that said attractor does not conform to said set of data.

15 67. The system of claim 65 further comprising the step of:
means for modifying said data masking function in response to a determination that said attractor does not conform to said data set.

20 68. A method for generating a compressed representation of a data set, substantially as herein described with reference to the accompanying drawings.

69. A method for generating a transformed data set which represents a data set, substantially as herein described with reference to the accompanying drawings.

25 70. A system for generating an attractor from a data transformation function and a data masking function, substantially as herein described with reference to the accompanying drawings.

30 71. A system for generating a compressed representation of a data set, substantially as herein described with reference to the accompanying drawings.

72. A system for generating a transformed data set which represents a data set, substantially as herein described with reference to the accompanying drawings.

DATED this Fifth Day of November 1998

35 **Iterated Systems, Inc.**
Patent Attorneys for the Applicant
SPRUSON & FERGUSON

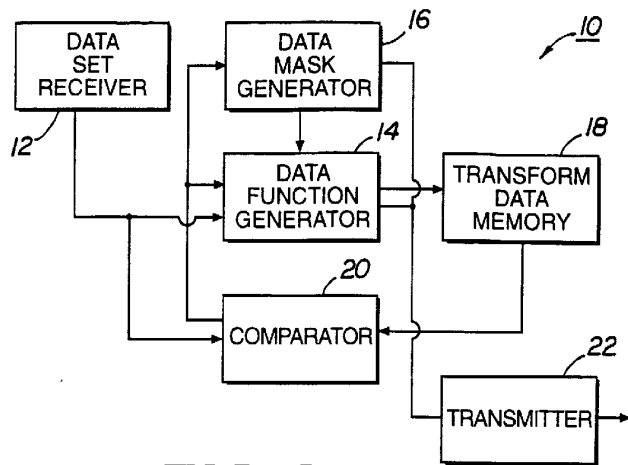


FIG 1

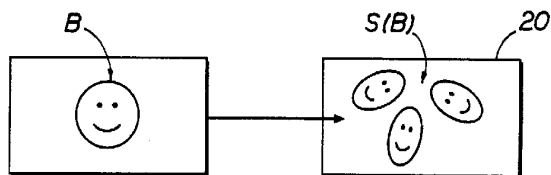


FIG 2

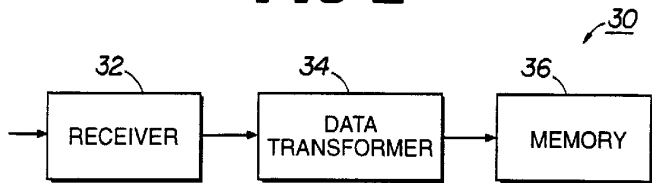


FIG 3

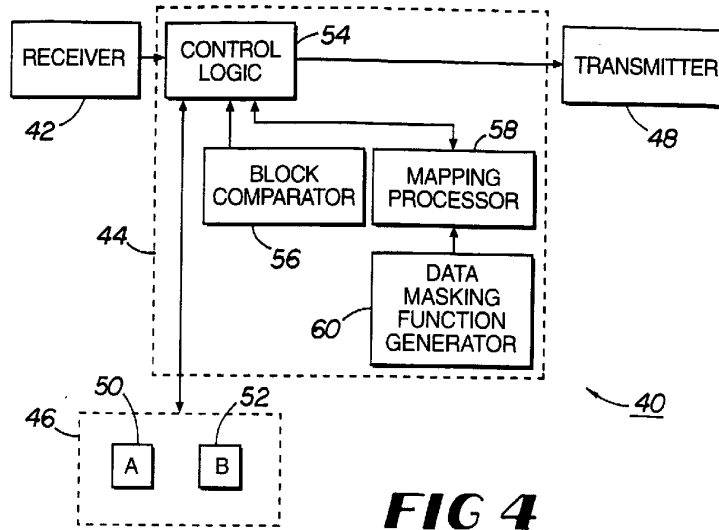


FIG 4

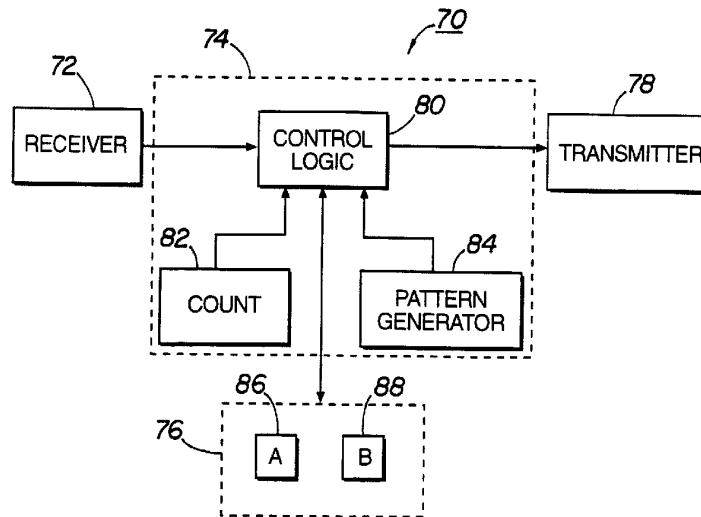


FIG 5

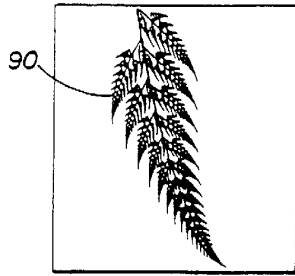


FIG 6

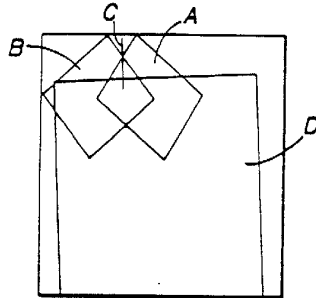


FIG 7



FIG 8

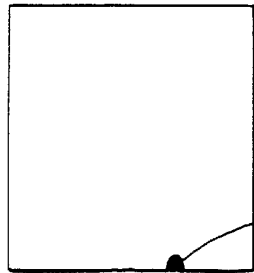


FIG 9

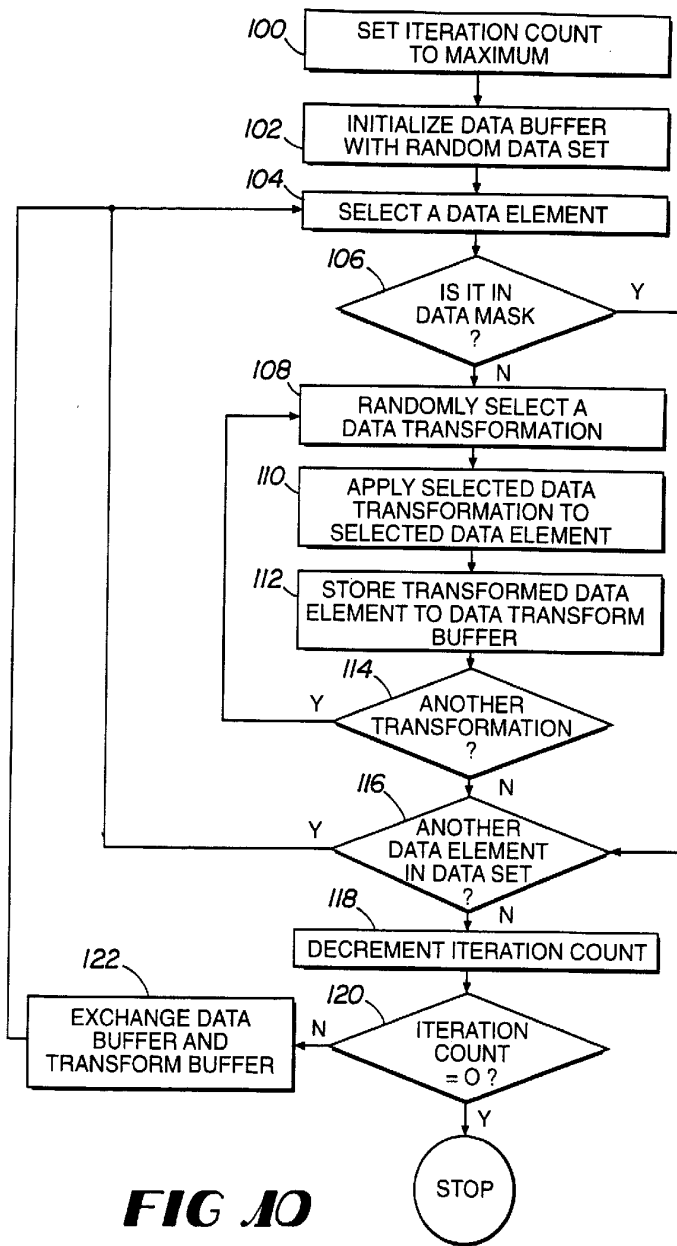


FIG 10

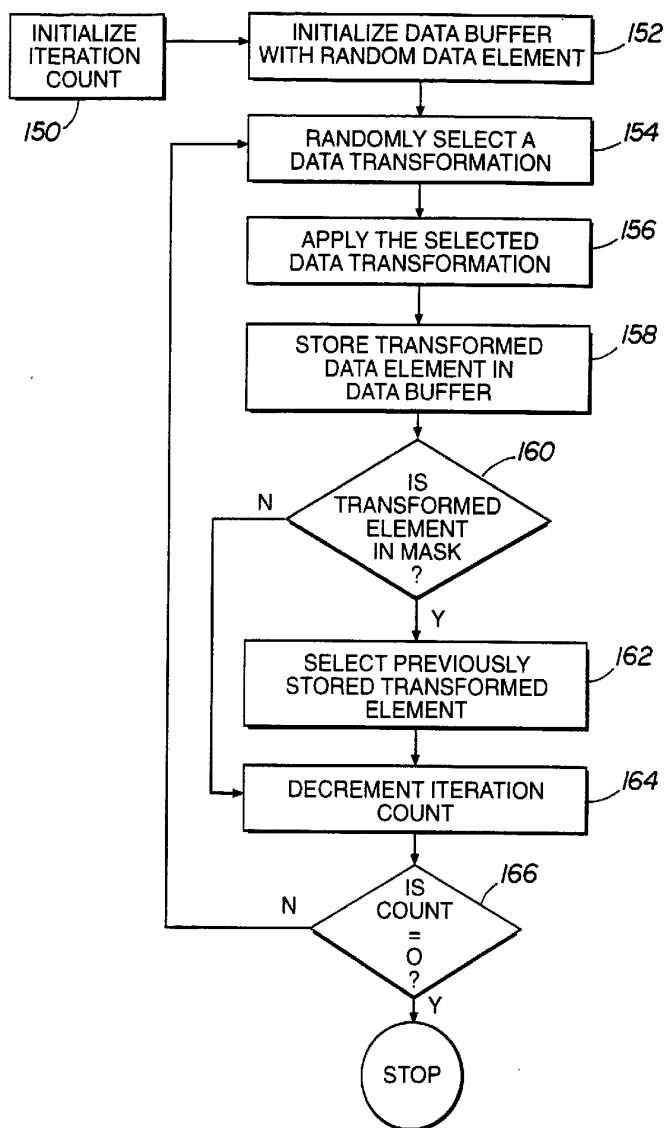


FIG 11

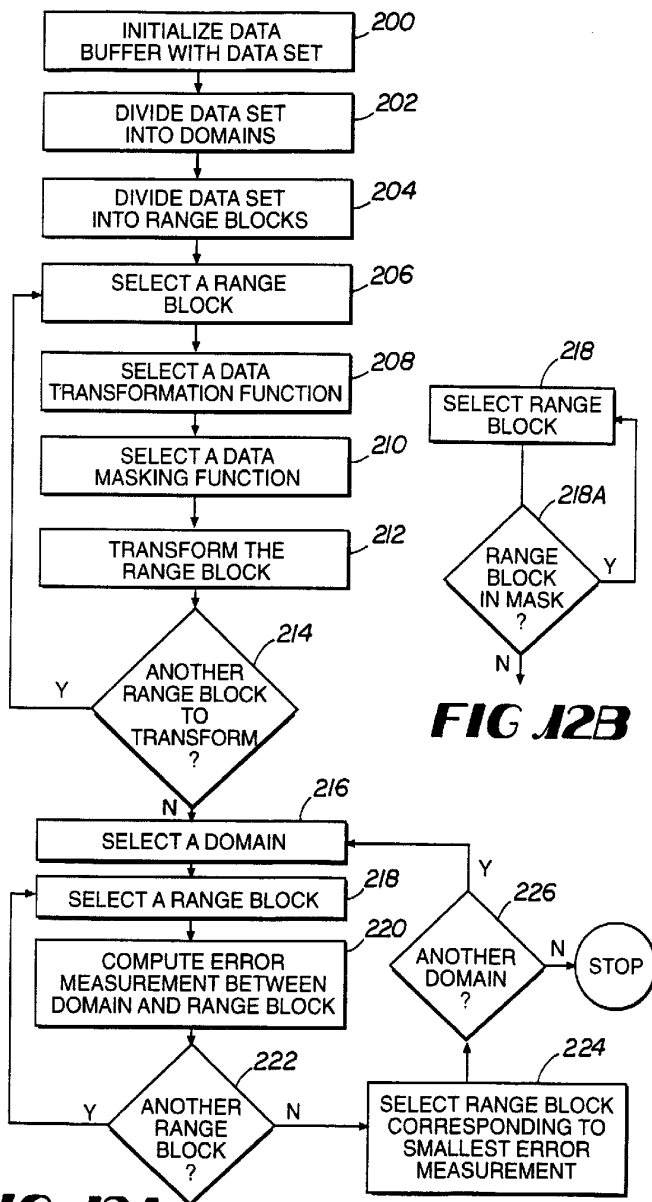


FIG 12B

FIG 12A