

Serial No. 641,018
Filing Date 15 April 1996
Inventor John M. Impagliazzo
Walter E. Greene
Quyen Q. Huynh

NOTICE

The above identified patent application is available for licensing. Requests for information should be addressed to:

OFFICE OF NAVAL RESEARCH
DEPARTMENT OF THE NAVY
CODE OCCC3
ARLINGTON VA 22217-5660

DISTRIBUTION STATEMENT A

**Approved for public release;
Distribution Unlimited**

19960712 068

DTIC QUALITY INSPECTED 1

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

2
3 SYSTEM AND METHOD FOR DATA COMPRESSION

4
5 STATEMENT OF GOVERNMENT INTEREST

6 The invention described herein may be manufactured and used
7 by or for the Government of the United States of America for
8 governmental purposes without the payment of any royalties
9 thereon or therefor.

10
11 BACKGROUND OF THE INVENTION

12 (1) Field of the Invention

13 The present invention relates in general to data compression
14 techniques and, more particularly, to a system and method for
15 wavelet-based data compression for digital transmission or
16 storage of images or signals.

17 (2) Description of the Prior Art

18 Data compression systems are essential to meet the digital
19 storage and transmission demands of these current and evolving
20 image and signal processing applications. Current image
21 compression techniques such as the standard Joint Photographic
22 Expert Group (JPEG) techniques are unable to meet the increasing
23 demands for higher compression ratios and greater image fidelity
24 needed by many applications.

25 The JPEG technique is based on the discrete cosine
26 transform. The technique divides the image into a series of 8x8
27 pixel blocks and the discrete cosine transform is computed for

1 each block producing an average value and 63 frequency values.
2 These values can be quantized to produce a minimum set
3 representing each 8x8 block. Since at least one value is needed
4 for each block, the highest compression ratio achievable is 64.
5 Additionally, because the image is divided into 8x8 blocks, the
6 approach tends to mask details, interrupt continuous lines and
7 cause discontinuities at the borders thereby lowering image
8 fidelity in the reconstructed image.

9 Approaches designed to increase the compression ratio and
10 increase image fidelity have focused on using wavelet based
11 applications. It is known that wavelet based applications can be
12 used to efficiently compress images as well as improve noisy
13 images. For example, United States Letters Patent Nos.
14 5,315,670, 5,321,776 and 5,412,741 to Shapiro describe a
15 technique of compressing an image by coding the addresses of the
16 threshold crossings of a significance map. This technique,
17 described as the zerotree structure, uses a symbol to identify
18 root coefficients as those coefficients having no descendant
19 coefficients with a magnitude greater than a given level. United
20 States Letters Patent 5,453,945 to Tucker et al. describes a
21 technique of employing an adaptive wavelet transform to generate
22 a set of wavelet coefficients having a lowest bit measure.

23 However, wavelet based techniques often suffer from many
24 disadvantages as well. For example, the technique of the Shapiro
25 patents is computationally intensive and complex to implement.
26 The technique of Tucker et al. requires a time-consuming and
27 computationally complex set of steps to determine the wavelet

1 basis or wavelet packet which provides the greatest degree of
2 compression for a particular signal. For a 512 x 512 image, this
3 technique could require 1024 bases, a different basis for each
4 row and column. Tracking the bases increases the amount of data
5 required in the compressed image.

6 Still other applications require such a large amount of data
7 to describe and code the addresses of the selected coefficients
8 that the compression benefits of the wavelet transforms are lost.

9 Thus, what is needed is an image compression system which
10 can reliably and efficiently generate a compressed image having a
11 high compression ratio and maintaining image fidelity.

12 13 SUMMARY OF THE INVENTION

14 Accordingly, it is a general purpose and object of the
15 present invention to provide an improved image compression system
16 and method.

17 A further object of the present invention is the provision
18 of a wavelet based image compression system which is less
19 computationally complex than prior image compression systems.

20 It is another object of the present invention to provide a
21 wavelet based image compression system having a simple and
22 efficient method of selecting coefficients for retention.

23 Yet a further object of the present invention to provide a
24 wavelet based image compression system which is computationally
25 efficient and requires less data to describe the addresses of the
26 selected coefficients.

1 These and other objects made apparent hereinafter are
2 accomplished with the present invention by providing an image
3 preprocessor for generating a digitized, formatted set of pixels.
4 An image transformer operates on the formatted set of pixels to
5 generate matrix of wavelet coefficients via a series of one-
6 dimensional wavelet transforms. A compression processor selects
7 coefficients for retention based on areas of interest in the
8 original image and the position of coefficients within the
9 coefficient matrix. The selection criteria can be varied for
10 each image thereby allowing for retaining greater image fidelity
11 in the compressed image. The processor also builds a compressed
12 image file that uses a coefficient location identifier (CLI) to
13 reduce the size of the addresses needed to indicate the position
14 of the selected coefficients.

15 16 BRIEF DESCRIPTION OF THE DRAWINGS

17 A more complete understanding of the invention and many of
18 the attendant advantages thereto will be readily appreciated as
19 the same becomes better understood by reference to the following
20 detailed description when considered in conjunction with the
21 accompanying drawings wherein like reference numerals and symbols
22 designate identical or corresponding parts throughout the several
23 views and wherein:

24 FIG. 1 is a block diagram illustrating image compression and
25 image decompression systems in accordance with the present
26 invention;

1 FIG. 2 is a diagram illustrating the structure of a
2 coefficient matrix;

3 FIG. 3A is a diagram illustrating a region of interest
4 within an image matrix;

5 FIG. 3B shows the coefficient matrix for the region of
6 interest of FIG. 3B; and

7 FIG. 4 is a flowchart diagram illustrating a process for
8 generating a coefficient location identifier.

9 10 DESCRIPTION OF THE PREFERRED EMBODIMENT

11 Referring now to FIG. 1, there is shown an image compression
12 system 10 and an associated image reconstruction system 20 in
13 accordance with the present invention. Image preprocessor 12
14 receives an input signal or image I and generates a digitized,
15 formatted set of pixels appropriate for input to image
16 transformer 14. The formatted set of pixels is passed to
17 transformer 14 where it is transformed into a set of wavelet
18 coefficients via a series of one-dimensional wavelet transforms.
19 In generating wavelet coefficients, transformer 14 can obtain a
20 desired wavelet transform from a wavelet library 16 containing a
21 set of useful wavelet bases. Transformer 14 passes the set of
22 wavelet coefficients to compression processor 18, which selects
23 and compresses a subset of the wavelet coefficients thereby
24 generating a compressed image file I_c representation of the input
25 I suitable for digital transmission or storage.

26 Reconstruction system 20 operates on the compressed image
27 file I_c to generate reconstructed image I_R . Compressed image

1 file I_c is received by decompression processor 22 where the
2 subset of selected wavelet coefficients are restored from I_c .
3 Inverse image transformer 24 performs an inverse transform
4 operation, selecting the appropriate wavelet basis from wavelet
5 library 26, on the restored set coefficients to generate a
6 restored set of pixels from which image processor 28 can generate
7 reconstructed image I_R .

8 Input signal I can comprise a data stream generated by one
9 or more sensors, data from an imaging device, an image previously
10 stored in a digital format or the like. In operation, image
11 preprocessor 12 receives signal I and generates an image array
12 comprising a two-dimensional $M \times N$ array of pixels $p_{m,n}$. For the
13 purposes of this discussion, it will be assumed that a single
14 image array with $M=N=512$ will be generated for each image and
15 that each pixel $p_{m,n}$ is assigned a value between 0 and 255
16 representing the gray level of that pixel. Although it will be
17 apparent to those skilled in the art that an image can be
18 comprised of more than one component array and that the pixels
19 can be distributed over any range. The procedure for generating
20 an image array from input signal I is dependent upon the form of
21 signal I and such procedures are well known in the art.

22 After generating the image array, preprocessor 12 reduces
23 the image array by subtracting the mean of the image from the
24 image. That is, each pixel $p_{m,n}$ is reduced as follows:

$$25 \quad p_{m,n} = p_{m,n} - \bar{I} \quad (1)$$

1 where $\bar{I}$ is the image mean and is given by:

$$2 \quad \bar{I} = \sum_{m=1}^M \sum_{n=1}^N \frac{p_{m,n}}{(M)(N)} . \quad (2)$$

3 The image mean is retained for incorporation in compressed
4 image file I_c . The mean is restored to the image by
5 reconstruction system 20. The image mean is subtracted from the
6 image to reduce the size of the wavelet coefficients generated by
7 transform processor 14.

8 The reduced image array, is passed to transform processor
9 14, where the reduced image array is transformed via a series of
10 one-dimensional wavelet transforms. The transform processing is
11 accomplished by first performing a one-dimensional transform on
12 each row of the reduced image and replacing the pixels $p_{m,n}$ with
13 the wavelet coefficients calculated. After all of the rows are
14 transformed, a one-dimensional wavelet transform is performed on
15 each of the columns, which now hold the coefficients from the row
16 transforms. The result of the two one-dimensional transforms is
17 a coefficient matrix which is the same size as the image matrix.
18 That is, transformer 14 generates an $M \times N$ matrix of wavelet
19 coefficients $c_{m,n}$, herein referred to as a coefficient matrix.

20 In generating wavelet coefficients, processor 14 is able to
21 select a wavelet basis from wavelet library 16 and, if desired,
22 processor 14 can select a different wavelet basis for each row
23 and column transformed. Therefore, in addition to generating the
24 wavelet coefficient matrix, transformer 14 also generates a list
25 indicating the basis used with each row and column. Using only

1 one row basis and one column basis will reduce the amount of
2 information which must be included in image file I_c since only
3 the two bases used need to be stored. Furthermore, if one row
4 basis and one column basis can be agreed upon *a priori*, then these
5 bases can be stored in the library and need not be included in
6 image file I_c , thereby further reducing the amount of data
7 contained in I_c .

8 It may be advantageous in some applications to select a
9 different basis to be used for each row or for sets of rows as
10 well as for each column or set of columns. However, storing the
11 bases for all M rows of the image would require as much storage
12 as the image itself. An entire basis for a single row requires a
13 number for each element in a row (N numbers are required for the
14 transform for each row m). Therefore, in a preferred embodiment
15 a set of L useful bases, each being uniquely identified such as
16 by a number 1 through L , is created and maintained in wavelet
17 library 16. Wavelet library 26 contains an identical set of
18 bases. With this arrangement, the basis used to transform a row
19 or column can be identified by its number, thereby requiring only
20 $M+N$ numbers to be stored.

21 The coefficient matrix and, if applicable, a basis list
22 indicating the bases used to transform the rows and columns, are
23 passed to compression processor 18 which generates compressed
24 image file I_c . Due to the compression capabilities of the
25 wavelet transform, the coefficient matrix should contain very few

1 large coefficients with most of the coefficients $c_{m,n}$ being zero.
2 By retaining only the largest coefficients, high compression is
3 possible with very little loss of information or fidelity in the
4 reconstructed image. Processor 18 generates image file I_c by
5 selecting a subset of the non-zero coefficients and storing the
6 selected coefficients along with a location identifier that
7 identifies the row address and the column address for each
8 selected coefficient.

9 The criterion used in selecting coefficients for retention
10 may be tailored to meet specific application requirements. In
11 some applications image fidelity must be maintained, but the
12 compression ratio is the primary concern. For example, a minimum
13 image compression ratio may need to be maintained to keep pace
14 with a real-time data acquisition rate or to satisfy the
15 bandwidth limitations of a communications link used to transfer
16 the images from an acquisition system to a remote platform for
17 analysis. Conversely, in many applications a high compression
18 ratio is needed, however image fidelity is the primary concern.

19 Similarly, in image recognition or classification
20 applications, an image may be obtained in a remote location and
21 transmitted to another location for processing, or the image may
22 be saved in a digital format for processing at a later time.
23 Thus, both compression ratio and image fidelity are of concern;
24 however, the compressed image must maintain the features from the
25 original image which are used for classification.

1 Several techniques can be used to evaluate and select
2 coefficients for retention. One technique is to select the X
3 largest coefficients where X is the number of coefficients
4 allocated for the compressed image file I_c . By largest
5 coefficients it is meant the coefficients of largest magnitude.
6 Alternatively, processor 18 can select a subset of the non-zero
7 coefficients based on regions of interest in the original image
8 and/or the position of coefficients within the coefficient
9 matrix. To illustrate this selection technique, the structure of
10 the coefficient matrix and its use in identifying and evaluating
11 coefficients to be included in I_c will be described below.

12 Referring now to FIG. 2, there is shown a diagram
13 illustrating the structure of a coefficient matrix 30 as it
14 relates to the location of the coefficients $c_{m,n}$. Matrix 30 is an
15 exemplary coefficient matrix such as would be generated by
16 transformer 14 using the Daubechies Fast Wavelet Transform (FWT).
17 The horizontal and vertical correlation coefficients with the
18 Daubechies Fast Wavelet Transform for each level are located in
19 regions as shown in FIG. 2. The coefficients $c_{m,n}$ are organized
20 by level, where level zero can be thought of as the original
21 data. Level horizontal one (H1) is formed when the FWT is
22 applied to a row; the correlation coefficients for the largest
23 scale and, therefore, the finest resolution in position are
24 stored in the right half of matrix 30. Levels H2 and higher are
25 in the left half columns of matrix 30. Similarly, the column
26 coefficients are computed from intermediate results of

1 transforming the rows and are designated $V_1, V_2, \dots$, with level
2 V_1 stored in the lower half of matrix 30 and levels V_2 and higher
3 stored in the upper half.

4 A number of discrete regions within matrix 30 have been
5 identified by a horizontal and vertical level number, e.g., H_3V_1 ,
6 to illustrate the level structure within matrix 30. That is,
7 region 32 of matrix 30 corresponds to horizontal level one and
8 vertical level one (H_1V_1). Similarly, region 34 corresponds to
9 level H_1V_2 and region 36 which corresponds to level H_2V_1 . In
10 signal processing terms, the lower levels, such as region 32
11 correspond to high resolution in time and a low resolution in
12 frequency. The higher levels, such as region 38 (H_3V_3),
13 correspond to a higher resolution in frequency with lower
14 resolution in time.

15 Within each region or level, the coefficients are positioned
16 relative to where the information exists in the original image.
17 How the information is mapped into a coefficient matrix is shown
18 in more detail in FIGS. 3A and 3B. FIG. 3A shows an image matrix
19 40 with a region of interest within the image identified as 41.
20 FIG. 3B shows the coefficient matrix 30' generated by transformer
21 14 for region 41 of image matrix 40. As can be seen in FIG. 3B,
22 coefficient matrix 30' can be divided into several discrete
23 regions each of which can be identified by a unique level number.
24 For example, region 32' corresponds to H_1V_1 , region 34'
25 corresponds to H_1V_2 and region 36' corresponds to H_2V_1 . The non-
26 zero wavelet coefficients for the region of interest 41 of image

1 40 fill square blocks on the diagonal, with block 42 within
2 region 32' being four times larger than block 48 within region
3 39'. For regions of matrix 30' off the diagonal, blocks become
4 rectangular since fewer coefficients are required for one of the
5 dimensions as is illustrated with block 44 within region 34' and
6 block 44 within region 34'. It should be noted that the relative
7 position of the block containing the non-zero coefficients in
8 each HV level is consistent with where the region of interest is
9 located with respect to the overall original image.

10 The selection technique can use information regarding the HV
11 level in which a coefficient resides as well as where within that
12 level the coefficient is located to identify coefficients for
13 retention. Selecting coefficients based on regions of interest
14 in the original image and/or on the position (HV level) within
15 the coefficient matrix that the coefficients reside can ensure
16 that certain features of or areas in the original image will be
17 maintained in the compressed image and restored in the
18 reconstructed image.

19 Generally, the coefficients found in the higher levels will
20 be larger than the coefficients residing in the lower levels.
21 Thus, selecting the X largest coefficients (globally selecting
22 the coefficients having the largest magnitude) will often result
23 in most of the selected coefficients being from the higher HV
24 levels. While such a technique can be used to obtain a high
25 compression ratio and maintain image fidelity, it may eliminate
26 some irregularities or discontinuities in the original image

1 which are necessary for classification. To ensure the selection
2 of coefficients from various levels processor 18 can use a
3 selection function having a threshold that varies with the
4 location of the coefficient within said coefficient matrix. That
5 is, the threshold varies with the HV level. For example,
6 processor 18 may use a local maximum selection function which for
7 high HV levels select all coefficients which have a magnitude of
8 at least sixty percent of the mean coefficient magnitude for that
9 level. At lower HV levels the selection function would keep only
10 those coefficients having a magnitude greater than 1.5 times the
11 mean coefficient magnitude. The threshold value for each level
12 can be determined statistically such that at higher levels a
13 majority of the coefficients are kept while at lower levels only
14 those coefficients having a magnitude which varies significantly
15 from the mean are retained.

16 Alternatively, if a region of interest in the original image
17 can be identified, this information can be used to ensure that
18 fidelity is maintained in that area. For example, in many
19 current medical applications one or more images are generated in
20 one location and viewed by specialists for diagnosis in one or
21 more remote locations. In such applications, the original images
22 may be viewed before transmission allowing one or more regions of
23 interest to be selected. In transmitting the images, an over all
24 image fidelity can be maintained with a greater fidelity being
25 maintained in the regions of interest.

26 If one or more regions of interest in the original image can
27 be identified before compression, processor 18 may select

1 coefficients based on a selection function that weights
2 coefficients corresponding to region of interest in the original
3 image higher than coefficients outside of that region. Processor
4 18 would then select the coefficients having a selection function
5 output that exceeds a threshold value. This threshold value can
6 vary with the region or level of the coefficient matrix. For
7 example, coefficients which are positioned within the matrix
8 relative to where the region of interest in the original image
9 exists are weighted higher by multiplying their magnitude by a
10 weighting factor which is greater than one. Processor then
11 selects coefficients having a weighted magnitude which is greater
12 than some threshold value which can vary with the region or level
13 of the coefficient matrix. The threshold value can be equal to
14 or based on the mean of the unweighted magnitudes of the
15 coefficients within a region of interest or HV level.

16 Having selected the subset of coefficients from the
17 coefficient matrix to be retained, processor 18 generates the
18 compressed image file I_c . Image file I_c contains three basic
19 components: a header, coefficient location identifiers, and the
20 coefficients themselves. The header contains basic information
21 about the image, the processing performed by transformer 14.
22 That is, the header contains the information needed to perform
23 inverse transform processing including: the size of the original
24 image (number of rows and columns), the mean value of the
25 original image ($\bar{I}$), any multiplier used to scale the 8-bit or
26 16-bit coefficients selected, the number of coefficients

1 selected, the bases used to transform the rows and columns, and
2 the length of the wavelet filter used for the rows and for the
3 columns.

4 The coefficient location identifiers indicate the position
5 (row and column address) within the coefficient matrix of each
6 selected coefficient. As should be apparent from the above
7 discussion, the position of the coefficients within the
8 coefficient matrix is very important. To identify the position
9 within the matrix, a single address is found for each coefficient
10 by forming one vector of coefficients from the matrix. The
11 location of each coefficient can then be identified using a
12 single vector address. To reduce the size of the addresses, a
13 coefficient location identifier (CLI) is generated for each
14 retained coefficient. The CLI identifies the vector address of a
15 coefficient by storing the address as offsets from the start of a
16 page.

17 Each CLI comprises one or more offset bytes having n -bits.
18 The first $n-1$ bits indicate an offset value and the n^{th} bit
19 indicates whether the offset value is an address on the current
20 page or a jump to a new page. Using an offset byte comprising 8
21 bits allows a maximum page length of 128, as only 7 bits are
22 available to indicate an address on the current page with the
23 eighth bit set for example, to zero to indicate the offset is on
24 the current page or to one to indicate the offset is a jump to a
25 new page.

26 For example, three coefficients, residing in the 54^{th} , 98^{th} ,
27 and 400^{th} positions of the coefficient vector, can be identified

1 as follows. The location of the first coefficient requires a CLI
2 with one offset byte having the eighth bit set to zero and bits
3 1-7 set to 54. To identify the location of the second
4 coefficient also requires one offset byte as it is still on the
5 first page. Bit 8 is again set to zero and bits 1-7 are set to
6 98. To identify the third coefficient requires a CLI having two
7 offset bytes. The first byte, indicating a page jump, has the
8 eighth bit set to one and bits 1-7 set to 3, thereby identifying
9 a jump of three pages. The second byte, indicating an address on
10 the third page, has bit 8 set to zero and bits 1-7 set to 16.
11 The fourth page identifies positions 384 to 511. If the
12 difference of vector address between two coefficients requires a
13 jump of more than 127 pages, then two or more bytes indicating
14 page jumps are followed by a byte indicating the address on the
15 page are used as the CLI.

16 Referring now to FIG. 4, there is shown a flow chart of a
17 process for generating coefficient location identifiers in
18 accordance with the present invention. Step 100 resets the CLI
19 to default to a page jump of zero pages (offset value equal to
20 zero) and obtains the vector address of next coefficient to be
21 saved.

22 Step 102 determines whether the vector address of the
23 coefficient is on the current page, that is if the vector address
24 is less than or equal to the current page base address plus the
25 page length. If the vector address of the coefficient is not on
26 the current page, step 104 determines if the maximum offset value
27 has been reached. Assuming an 8-bit offset byte, step 104

1 determines if the offset value (bits 1-7) is less than 127. If
2 the offset value is less than 127, step 106 increases the offset
3 value, increases the current base page address to the base
4 address of the next page and returns to step 102.

5 If the step 104 determines that the maximum offset value has
6 been reached, step 108 formats and writes an offset byte which
7 indicates a page jump with a maximum offset value to the
8 compressed image file I_c . Step 108 also creates a new offset
9 byte having a page jump of one page and returns to step 102.

10 If step 102 determines that the vector address of the
11 coefficient is on the current page, then step 110 writes the CLI
12 to the compressed image file I_c . If the current offset value is
13 equal to zero (no page jumps) then step 110 determines the offset
14 value for the location of the coefficient on the current page and
15 writes this offset byte to I_c . If the offset value is greater
16 than or equal to one (indicating a page jump), step 110 first
17 writes the offset byte for the page jump and then writes the
18 offset byte for the location of the coefficient on the current
19 page. Step 110 can determine the offset value for the location
20 of the coefficient by subtracting the current page base address
21 from the vector address of the coefficient.

22 The process is repeated, beginning with step 102, for each
23 coefficient selected until step 112 ascertains that a CLI has
24 been generated for each selected coefficient.

25 The final component of compressed image file I_c contains the
26 selected coefficients. To reduce the size of I_c , the
27 coefficients are scaled such that they can be represented using a

1 predetermined number of bits. For example, the coefficients can
2 be scaled to fit into 16 or 8 bit integers. To scale the
3 coefficients, the magnitude of the largest coefficient is found.
4 If the magnitude of the largest coefficient is between -128 and
5 127 (an 8 bit number), then all the coefficients are scaled up by
6 a multiplier so that the largest coefficient is equal to 127 (or
7 -128 if negative). The coefficients are then rounded to the
8 nearest integer and stored as 8-bit integers. If the magnitude
9 of the largest coefficient is between 127 and 32,767 (or between
10 -128 and -32,768 if negative), the coefficients are rounded to
11 the nearest integer and stored as 8-bit integers or 16-bit
12 integers as necessary. If the magnitude of the largest
13 coefficient greater than 32,767 (or smaller than -32,768), then
14 all the coefficients are scaled down by a multiplier such that
15 the largest coefficient is equal to 32,767 (or -32,768 if
16 negative). The coefficients are then rounded to the nearest
17 integer and stored as 8-bit or 16-bit integers as necessary.

18 In generating the compressed image file I_c , processor 18
19 first formats and builds the header. For each coefficient
20 selected, processor 18 then generates and writes a CLI followed
21 by the corresponding coefficient. The structure of a compressed
22 image file is illustrated in Table 1.

TABLE 1

Component	Item	Description
Header	Image Mean	Mean value of the original image ($\bar{I}$)
Header	Scaling Multiplier	Multiplier used to scale the wavelet coefficients
Header	Rows	Number of rows in the original image
Header	Columns	Number of columns in the original image
Header	Row Basis	Lists the bases used to transform the rows
Header	Column Basis	Lists the bases used to transform the columns
Header	Row Level	Contains level, if a level basis is used to transform the row
Header	Column Level	Contains level, if a level basis is used to transform the column
Header	Row Filter Length	Wavelet filter length used for the rows
Header	Column Filter Length	Wavelet filter length used for the columns
Header	Coefficients kept	The total number of coefficients selected for retention
Header	8-bit coefficients kept	The number of 8-bit coefficients stored in the file
Header	16-bit coefficients kept	The number of 16-bit coefficients stored in the file
Coefficient Location	Location identifier	Location identifier for the following 8-bit coefficient
Coefficient	8-bit coefficient	8-bit coefficient
Coefficient Location	Location identifier	Location address for the following 16-bit coefficient
Coefficient	16-bit coefficient	16-bit coefficient

Reconstruction system 20 operates on the compressed image file I_c generated by image compression system 10 to build a reconstructed image I_R . Decompression processor 22 receives the compressed image file, extracts the header information from the file, and builds a reconstructed coefficient matrix. Processor 22 places the selected coefficients in the reconstructed

1 coefficient matrix using the CLI stored for each coefficient.
2 Processor 22 then fills in the remaining positions in the matrix
3 with zeros.

4 Inverse image transformer 24 reconstructs a restored image
5 matrix from the reconstructed coefficient matrix via a series of
6 inverse wavelet transforms. In generating the restored image
7 matrix, transformer 24 is able to obtain representative wavelet
8 bases from library 26 in accordance with known principals. The
9 restored image matrix is passed to processor 28 which restores
10 the image mean to each pixel and combines any component image
11 matrices into a reconstructed image.

12 What has thus been described is an image compression system
13 and an associated reconstruction system. These systems offer
14 significant advantages over the prior art. The compression
15 system obtains high compression ratios by substantially reducing
16 the addressing information needed to identify the location of
17 selected coefficients. Additionally, the compression system
18 provides an efficient means for selecting coefficients.

19 It will be understood that various changes in the details,
20 materials, steps and arrangement of parts, which have been herein
21 described and illustrated in order to explain the nature of the
22 invention, may be made by those skilled in the art within the
23 principle and scope of the invention.

2
3 SYSTEM AND METHOD FOR DATA COMPRESSION

4
5 ABSTRACT OF THE DISCLOSURE

6 A data compression system includes an image preprocessor for
7 generating a digitized, formatted set of pixels which is passed
8 to an image transformer. The image transformer generates a set
9 of wavelet coefficients from the formatted set of pixels via a
10 series of one-dimensional wavelet transforms. A compression
11 processor selects a subset of the wavelet coefficients for
12 retention based on areas of interest in the original image and
13 the position of coefficients within the set of coefficients. The
14 compression processor then builds a compressed image file using a
15 coefficient location identifier to reduce the size of the
16 addresses that indicate the position of the selected coefficients
17 within the set.

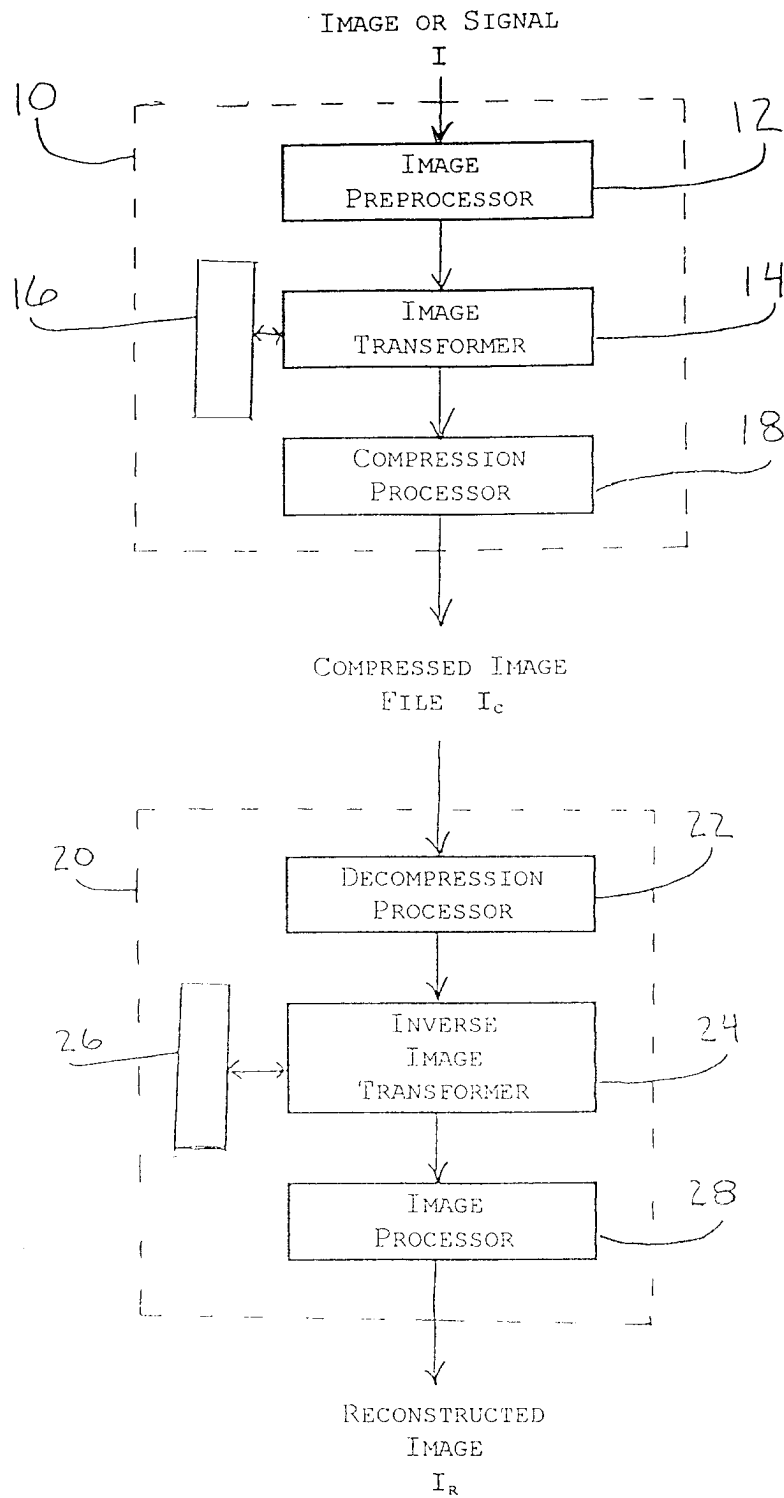


FIG. 1

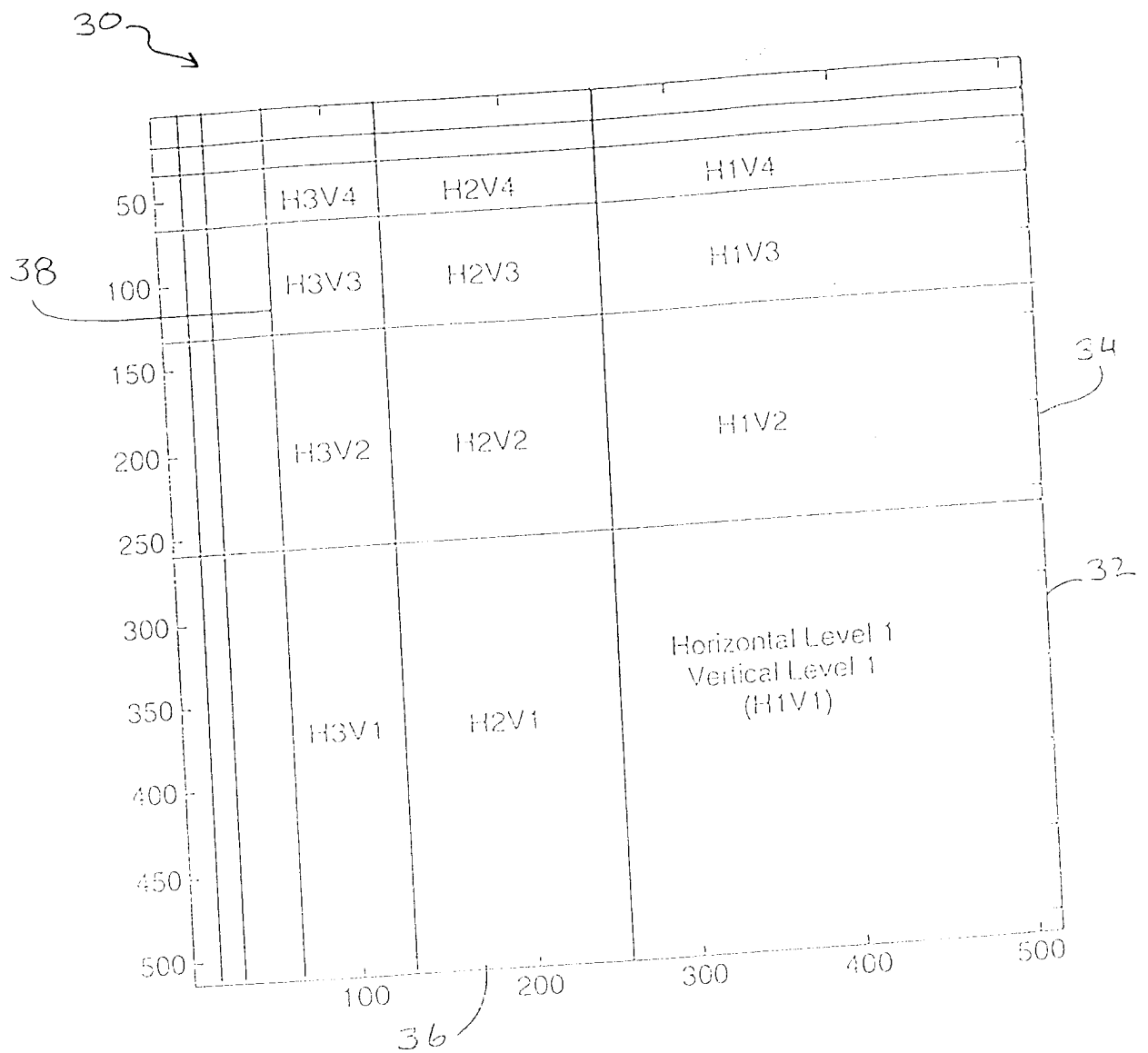


FIG. 2

40
→

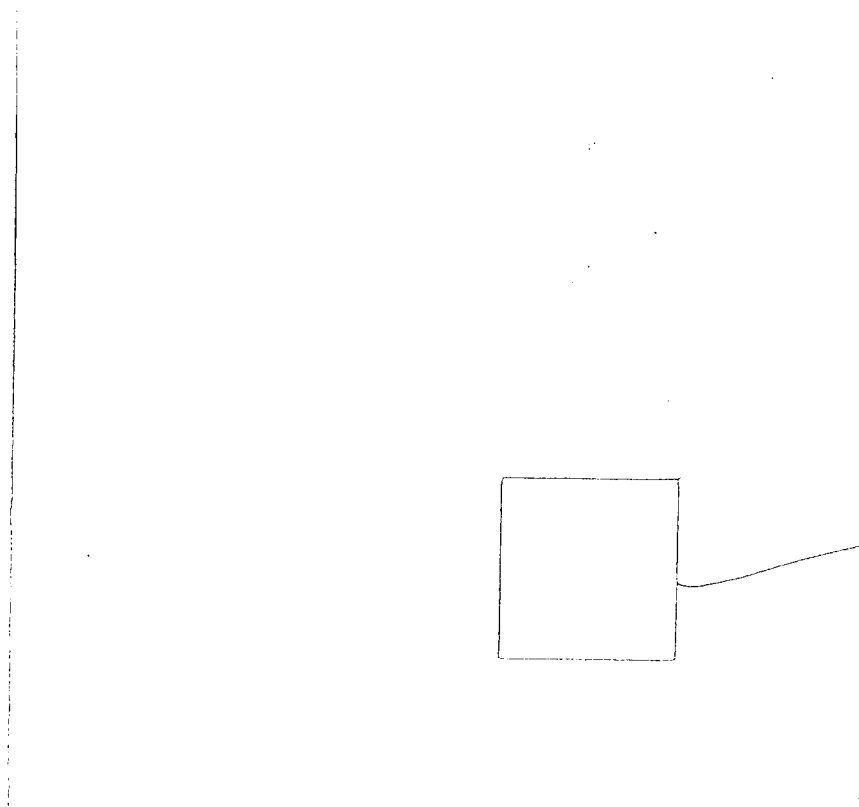


FIG. 3A

30'
→

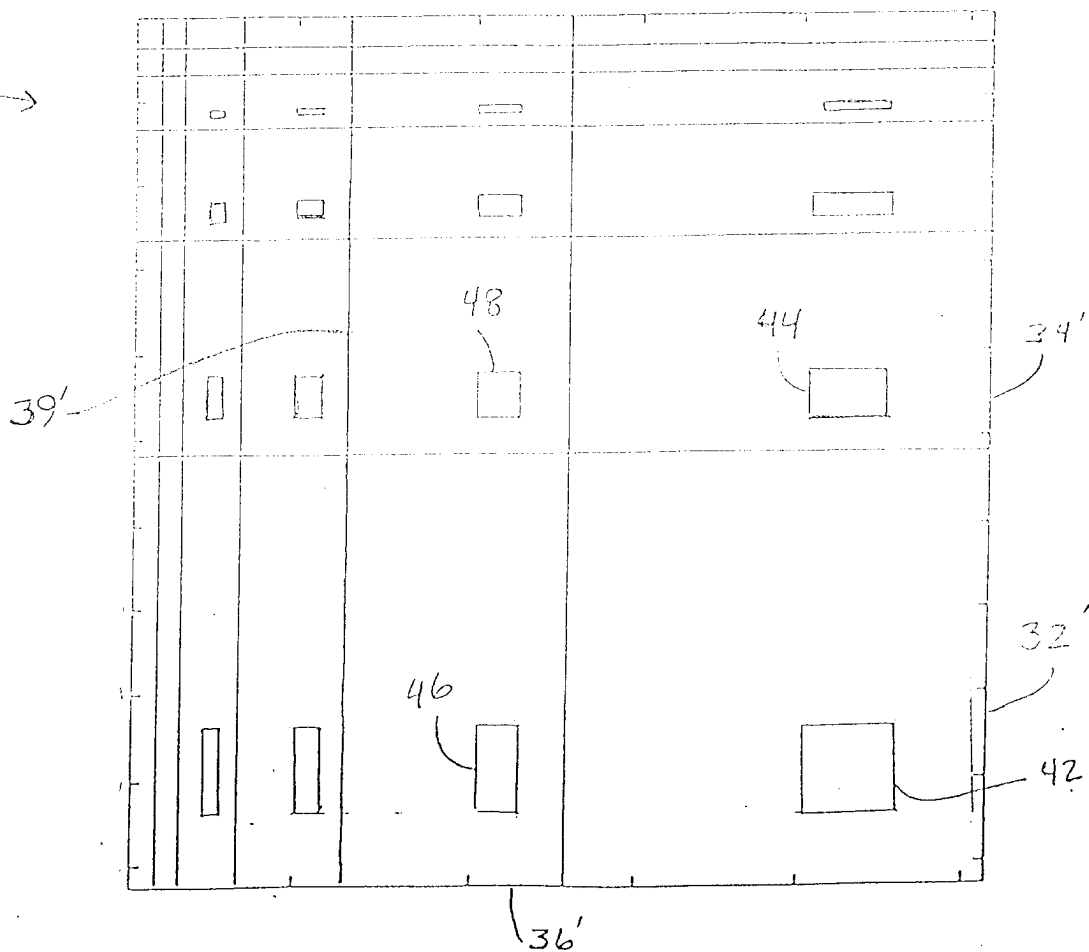


FIG. 3B

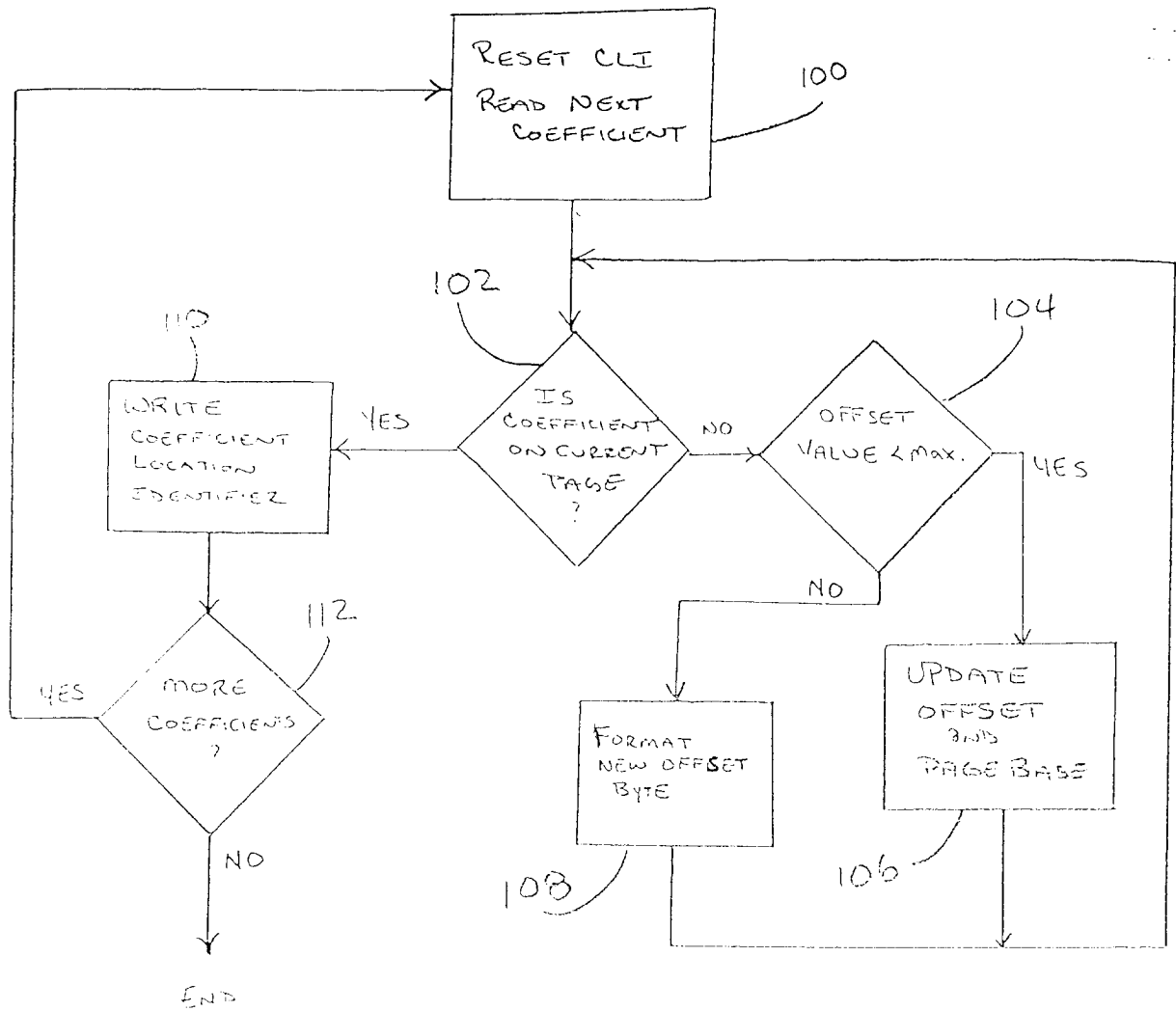


FIG. 4.