

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
20 September 2007 (20.09.2007)

PCT

(10) International Publication Number
WO 2007/104522 A1

(51) International Patent Classification:

G06T 9/00 (2006.01) **H04N 7/30** (2006.01)
H04N 7/26 (2006.01) **H04N 1/417** (2006.01)

(21) International Application Number:

PCT/EP2007/002158

(22) International Filing Date: 12 March 2007 (12.03.2007)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

06004986.3 10 March 2006 (10.03.2006) EP

(71) Applicant (for all designated States except US): **MI-CRONAS GMBH** [DE/DE]; Hans-Bunte-Strasse 19, 79108 Freiburg (DE).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **RIEDER, Peter** [DE/DE]; Putzbrunnerstr. 9c, 81737 München (DE). **SUNG, Chih-Ta Star**; Rm. 308, Build.52, N°195, Sec.4, Chung-Hsing Rd., Chu Tung, HsinChu (TW). **LAN, Yin-Chun Blue**; Rm. 308, Build.52, N°195, Sec.4, Chung-Hsing Rd., ChuTung, HsinChu (TW).

(74) Agent: **BICKEL, Michael**; Westphal, Mussnug & Partner, Mozartstrasse 8, 80336 Muenchen (DE).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

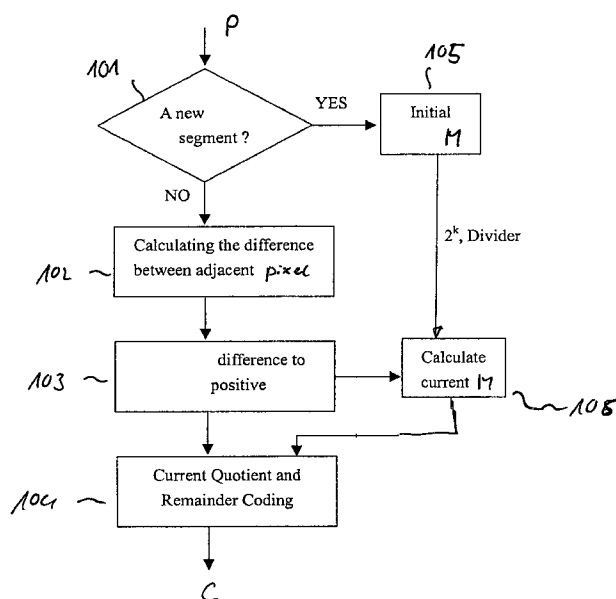
(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— with international search report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: METHOD AND APPARATUS FOR LINEWISE IMAGE COMPRESSION



(57) Abstract: Disclosed is a method and an apparatus for linewise image compression. The method comprises: subdividing the lines into segments, with each of the segments comprising a number of pixels, with each pixel being assigned at least one pixel value; assigning a bit budget to each segment; compressing the segments for obtaining compressed segments using one of at least two coding methods having different coding qualities, with the coding method being selected such that a bit number of the compressed segment is equal to or lower than the bit budget; wherein the bit budget of a segment of one line is selected dependent on a coding quality obtained for an adjacent segment in an adjacent line.

METHOD AND APPARATUS FOR LINEWISE IMAGE COMPRESSION

The invention relates to a method and an apparatus for linewise image compression.

5

BACKGROUND OF THE INVENTION

Images of video sequences comprise a number of pixels arranged in lines, with each of these pixels being assigned at least one pixel values. In order to reduce memory space and/or memory bandwidth required for storing such images compression of the pixel data is desired.

10

SUMMARY OF THE INVENTION

A method for compressing an image having lines of pixels according to one example of the invention comprises: subdividing the lines into segments, with each of the segments comprising a number of pixels; assigning a bit budget to each segment; compressing the segments for obtaining compressed segments using one of at least two coding methods, with the coding method being selected such that a bit number of the compressed segment is equal to or lower than the bit budget; wherein the bit budget of a segment of one line is selected dependent on a coding quality obtained for an adjacent segment in an adjacent line.

15

20

25

30

An apparatus for compressing image data of an image having lines of pixels according to an example of the invention comprises: means for subdividing the lines into segments, with each of the segments comprising a number of pixels, with each pixel being assigned at least one pixel value; means for assigning a bit budget to each segment; means for compressing the segments for obtaining compressed segments using one of at least two coding methods, with the coding method being selected such that a bit number of the compressed segment is equal to or lower than the bit budget; means for selecting the bit budget of a segment of one line dependent on a coding quality obtained for an adjacent segment in an adjacent line.

BRIEF DESCRIPTION OF THE DRAWINGS

Examples of the present invention will be explained in the following with reference to the enclosed drawings.

- 5 Figure 1 illustrates two methods for linewise image compression, wherein a first method applies a fixed compression rate, while a second method applies a variable compression rate.
- Figure 2 illustrates subdividing lines of an image into segments.
- 10 Figure 3 illustrates the distribution of differences between adjacent pixels in a segment according to an example of the invention.
- Figure 4 illustrates mapping differences between pixel values to positive values.
- 15 Figure 5 illustrates generating a variable length code.
- Figure 6 shows a table including examples of variable length codes.
- Figure 7 illustrates a method for calculating variable length codes.
- Figure 8 shows an example for adapting bit budgets assigned to line segments.

DETAILED DESCRIPTION OF THE DRAWINGS

Figure 1A illustrates a first method for linewise image compression.

According to this method a line 1 of an image is subdivided in a number of segments $1_1, 1_2, \dots, 1_n$, wherein each segments includes a number of adjacent pixels $11_1, 11_2$. A pixel value is assigned to each pixel. The pixel values may either be luminance values (Y) or chrominance values (U, V). The pixel values are represented by binary words, for example, binary words having a bit length of 8 bit. Applying a fixed compression rate the binary words representing the pixel values may be compressed, thereby obtaining compressed bit sequences $C1_1, C1_2, \dots, C1_n$ with the overall number of bits of these sequences being shorter than the overall number of bits of one segment. Compression may, for example, be achieved by truncating a number of least significant bits from the binary words representing the pixel values.

Figure 1B illustrates a compressing method using a variable compression rate. The ratio between the number of bits of the compressed sequences and the number of bits of the segments varies with this method, with this ratio corresponding to the compression rate. A compression rate varying on a segment by segment basis may be achieved using variable length coding (VLC).

An example of a line compression method according to the invention will be explained in the following with reference to Figure 2. In this method every line of an image comprising a number of lines 1, 2, 3, ... is subdivided into segments $1_k, 2_k, 3_k, \dots$. Each of these segments comprises a number of pixels, with at least one pixel value being assigned to each of these pixels. In the method segments of one line are compressed, whereas Information obtained on the complexity of image structures represented by pixel data of line are used for compressing segments of adjacent/neighbor lines, which will be explained in the following.

Instead of directly compressing the pixel values differences between the same type of pixel values of adjacent pixels are calculated and the differences

are compressed. This makes use of the fact, that adjacent pixels of one line are often correlated, which results in small differences, which may be compressed/encoded using short codes or code words. Let P_{n-1} and P_n be pixel values of two adjacent pixels of one segment, then a difference D_n to be coded is calculated as follows:

$$D_n = P_n - P_{n-1} \quad (1).$$

For illustration purposes it may be assumed that the pixel having pixel value P_{n-1} is a neighbor to the left of pixel having pixel value P_n . Encoding differences of pixel values, which in the following is denoted as differential encoding or DPCM encoding (DPCM = Differential Pulse Coded Modulation) is, of course, not possible for a first pixel of a first segment of one line. Such first pixel of the first segment in each line is encoded directly. A stream of pixel values to be decoded for the first segment therefore is: $P_0 D_1 D_2 D_3 \dots$, where P_0 denotes the pixel value of a first pixel of the segment, and $D_1, D_2, \dots$ denote differences. For first pixels of further segments of one line, the pixel value of a last pixel of a previous (left neighbor) segment is used as a reference value for calculating a first difference. A stream of pixel values to be decoded for further segments therefore is: $D_0 D_1 D_2 D_3 \dots$, where D_0 denotes a first difference calculated by $D_0 = P_0 - P_z$, where P_0 denotes the pixel value of a first pixel of a current segment, and P_z denotes the last pixel of a previous segment.

The pixel values P are represented by binary words. For illustration purposes the bit length of these binary words is assumed to be 8 bit. It goes without saying that the invention is not restricted to bit lengths of 8 bit, but any other bit lengths - even higher bit lengths as, for example 10 bit - may be used either. For bit lengths of 8 bit the pixel values P range between 0 and 255. Consequently, the differences D range between -255 and 255, i.e.

$$-255 \leq P_n - P_{n-1} \leq 255 \quad \text{where } 0 \leq P_{n-1} \leq 255 \text{ and } 0 \leq P_n \leq 255 \quad (2)$$

In general 9 bit would be required for coding such difference, and thereby allowing lossless decoding. However, assuming that one of the pixel values of the difference, namely P_{n-1} , is known, then the range for the difference D_n to be decoded can be narrowed to

5

$$-P_{n-1} \leq P_n - P_{n-1} \leq 255 - P_{n-1} \text{ where } 0 \leq P_n \leq 255 \quad (3),$$

which implies that the difference D_n may be encoded using only 8 bit.

10

For a correct reconstruction of D_n , if it is encoded using only 8 bit, it is required that P_{n-1} is known. For a first difference $D_1 = P_1 - P_0$ of a segment pixel value P_0 is coded directly, and is therefore available for decoding purposes. Knowing P_0 and the 8 bit data word representing the difference D_1 , the difference D_1 may be reconstructed. Using said reconstructed difference D_1

15

pixel value P_1 may be reconstructed, which may then be used for reconstructing a next difference $D_2 = P_2 - P_1$, etc. Starting from a first pixel value of segment differences being represented by 8 bit data words, and using these differences, further pixel values may be reconstructed by applying an iterative process.

20

According to an example of the invention the differences D_n prior to coding are mapped to values of a given interval including $2^8 = 256$ different values. It is obvious, that more than one difference D_n has to be mapped to the same value of the mapping interval if there are 2^9 different difference D_n and only 2^8 different values for representing those differences D_n . However, by

25 applying a mapping scheme that uses one of the pixel values, for example P_{n-1} , as a mapping parameter, the correct difference can be reconstructed from the mapped value using pixel value P_{n-1} .

30

Mapping a difference D_n to a value within the interval of 256 different values may be performed by simply shifting the difference D_n to a positive value using pixel value P_{n-1} as a shifting parameter.

Another approach for mapping the differences will be explained in the following:

Usually the pixel values of adjacent pixels in an image are correlated to each other. The differences between the pixel values of adjacent pixels, in a first approach, therefore are normally distributed, as depicted in Fig. 3. For effectively applying a variable length coding (VLC) to the differences D_n it is therefore desired to map small differences D_n to small values and to map larger differences D_n to larger values. A possible mapping scheme for mapping a difference D to a target value D' , with $0 \leq D' \leq 255$, using pixel value P_{n-1} as a mapping parameter is, for $P_{n-1} \leq 128$:

$$D_n' = P_{n-1} + D_n \quad \text{for } |D_n| > P_{n-1} \quad (4a)$$

$$D_n' = 2 \cdot |D_n| - 1 \quad \text{for } |D_n| \leq P_{n-1} \text{ and } D_n > 0 \quad (4b)$$

$$D_n' = 2 \cdot |D_n| \quad \text{for } |D_n| \leq P_{n-1} \text{ and } D_n < 0 \quad (4b)$$

$$D_n' = 0 \quad \text{for } D_n = 0 \quad (4c)$$

The mapping scheme according to Eqns. 4a to 4c maps differences in the range of $-P_{n-1} \leq D_n \leq P_{n-1}$ to target values between 0 and $2 \cdot P_{n-1}$, while differences D_n larger than P_{n-1} are shifted using P_{n-1} as a shifting parameter. Fig. 4A for $P_{n-1} = 3$ shows the distribution of possible differences (assuming that the differences are normally distributed) while FIG 4B shows the possible distribution of the target values D'_n obtained by mapping the differences. In the example $D=1$ is mapped to $D'=1$, $D=-1$ is mapped to $D'=2$, ..., $D=-3$ is mapped to $D'=5$ and $D=3$ is mapped to $D'=6$.

For $P_{n-1} > 128$ the same mapping scheme applies except for the mapping according to Eqn. 4a being replaced by

$$D' = -P_{n-1} + D_n \quad \text{for } |D_n| > P_{n-1} \quad (5a).$$

The target values D'_n , obtained by mapping the differences D_n to all positive values, may be encoded using variable length coding (VLC). For

assigning code words C to each of the target values D' these target values D' in a first step are expressed as

$$D' = M \cdot Q + R \quad (6),$$

5

where M denotes a divisor, Q denotes the integer result obtained by dividing D through divisor m, and R denotes the remainder of the division operation. For each target value D' in a second step integer Q and remainder R are mapped to a code word C as shown in Fig. 5. The code word C includes a number of first bits (zeros in the example), with this number corresponding to Q, a marker of a second bit (one in the example), and the binary equivalent of the remainder R. The bit length of the remainder R is x, with $2^{x-1} < M \leq 2^x$ or $x-1 < \log_2 M \leq x$.

10

Fig. 6 shows a table including several examples of codes obtained for different D values and different M values, where M is $2^3=8$, $2^4=16$ or $2^0=1$ in the example. Assuming that differences between the differences to be coded are small, and further assuming that M fits the differences to be coded well, short codes can be obtained.

15

20

According to an example of the VLC process divisor M is updated with each new target value D' to be encoded. Let M_{n-1} be the divisor for coding target value D'_{n-1} , a divisor M_n for coding a next target value D'_n may then be obtained by calculating a weighted sum of D'_{n-1} and M_{n-1} - for example the mean value $(D'_{n-1} + M_{n-1})/2$ - and rounding to the next 2^x , with x being an integer ≥ 0 , being equal to or smaller than the weighted sum. M_n may be expressed as :

25

$$M_n = (M_{n-1} + D_n) / 2 - [(M_{n-1} + D_n) / 2] \bmod 2^x \quad (7)$$

At the beginning of encoding pixels of a segment an initial value for divisor is selected. Such initial value may be a fixed value or may be adopted from coding the last difference of an adjacent segment.

30

Figure 7 is flowchart illustrating VLC coding process as discussed above. P in Figure 7 denotes a stream of pixel values. In a first step 101 a decision is made as to whether the incoming pixel values relate to a current segment or to a new segment. Under the assumption, that a fixed number of pixels is assigned to each segment, this decision may be realized using a counter that counts the incoming pixel values, whereas every time the counter reaches a given threshold, that represents the number of pixels per segment, the counter is reset and the beginning of a new segment is assumed. With every new segment divisor M is set to an initial value in step 102.

In step 102 the difference between pixel values of adjacent pixels are current, and these differences are mapped to positive target values in step 103. Based on these differences and based on divisor M, which may be the initial divisor or an updated divisor a current divisor is calculated in step 105. Such divisor is then used in step 104 for VLC coding the difference obtained in step 103. C in Figure 7 denotes a data stream of coded data words, with each data word representing a difference between two adjacent pixels.

In this connection it should be mentioned, that instead of coding differences between pixel values of adjacent pixels differences between a pixel value of a pixel and a prediction value of such pixel may be coded. The difference D_n would then read

$$P_n = P_n - E_n \quad (8),$$

where E_n is the prediction value. E_n according to fixed prediction scheme may be dependent on pixel values of several adjacent pixels. E_n may in particular be dependent on a number of left neighbor pixels, which allows the iterative reconstruction of differences and pixel values as discussed above:

$$E_n = f(P_{n-1}, P_{n-2}, P_{n-3}, \dots) \quad (9a),$$

where $f(.)$ denotes a weighting function. Such function may either be dependent on the pixel value P_n itself, i.e.

$$E_n = f(P_{n-1}, P_{n-2}, P_{n-3}, \dots) \quad (9b).$$

5

Prior to calculating the differences and therefore prior to applying a VLC method to the differences D or to the target values D' (in case the differences are mapped to target values) one or more least significant bits (LSB) may truncated from the binary words representing the pixel values.

10

Instead of applying a VLC method to differences between truncated or un-truncated pixel values the pixel values may be directly encoded by simply truncating one or more least significant bits. It goes without saying that truncating LSBs from the pixel values results to losses (lossy encoding).

15

An overall number B_C of bits for one segment of e.g. 32 pixels should be lower than the overall number B_P of bits of the binary words representing the pixel values in order to achieve a compression factor higher than 1, with the compression factor F being

20

$$F = B_P/B_C \quad (10).$$

25

For achieving a given compression factor a bit budget is assigned to each segment, whereas the bit budget denotes the overall number of bits the sequence of code words for the particular segment may have. Coding that meets the given bit budget may be performed lossless or non-lossless, which is dependent on the complexity of the image content represented by the pixels of one segment,. For finding a coding method that meets the bit budget test encoding is performed for each segment. Test encoding comprises encoding a given segment using different coding methods, whereas each difference or target value of one segment is encoded using the same method, and counting the overall number of bits of a code sequence obtained by the particular coding

30

method. Finally the segment is encoded using the coding method that meets the given bit budget best.

"Meeting the bit budget best" in this regard means, that the best coding
5 method is used the bit number of which is equal or lower than the bit budget. If lossless coding is possible, the lossless method is used that requires the lowest number of bits. If only lossy coding is possible, the best lossy method is used the bit number of which is equal to or lower than the bit budget. "Best coding method" in this regards means the coding method resulting to the lowest losses,
10 i.e. the least number of truncated bits, as compared to the other methods, while "worst coding method" in this regards means the coding method resulting to the highest losses.

The different coding methods available for coding the segments may be
15 ranked according to their losses, while "best coding method" in the following denotes the method resulting to the lowest losses, while "worst coding method" in the following denotes the method resulting to the highest losses.

It is a goal to achieve similar compression qualities for the segments of
20 one line. According to an embodiment of the invention the "worst segments" in one line are detected for which the worst coding method of the line is selected, and "best segments" in the same line are detected for which the best coding method is selected. "Worst coding method of the line" in this regard denotes the worst coding method of one line as compared to the other coding methods
25 in this line, while "best coding method of the line" in this regard denotes the best coding method of one line as compared to the other coding methods in this line.

The bit budget of segments adjacent to worst blocks in a next line is then increased as compared to the bit budget of the corresponding worst block, while
30 the bit budget of segments adjacent to best blocks in a next line is then decreased as compared to the corresponding best block. An example for this method is depicted in Figure 8, where the bit budget and the compression quality for segments of three adjacent lines 1, 2, 3 is depicted, where it is

assumed that five coding methods with four different coding qualities Q1...Q5 are available.

The segments of the first line 1 are compressed using a given same bit budget for each segment, which results to different coding qualities. According to the example a best coding quality is obtained for two blocks, and a worst coding quality is obtained for two blocks. The bit budget of segments adjacent to the worst blocks in a second line 2 is then increased, while the bit budget of segments adjacent to the best blocks in the second line 2 is then decreased.

In the example a best coding quality is obtained for three blocks in the second line, where in the example this best quality is worse than the best quality than the best coding quality obtained for the first line. A worst coding quality is obtained for three blocks in the second line. The bit budget for three segments adjacent to the three worst blocks in the third line is then increased, while the bit budget for the three best blocks is decreased. While for the first line the best coding has quality ranking Q5 and the worst coding has quality ranking Q1 the best and worst coding of the third line have rankings Q2 and Q3, i.e. the coding qualities have been made similar due to adapting the bit budget.

According to an embodiment, the bit budget is increased or decreased in same steps, e.g. 16 bit per segment. According to a further segment the number of segments for which the bit budget is increased from line to line is equal to the number of segments for which the bit budget decreased. This helps keeping the overall bit budget for one line equal from line to line.

Since the complexity of image structures represented by different segments of one line may vary, the coding methods applied to these segments may vary either, whereas some segments may be coded lossless, while other segments may be coded lossy. According to an embodiment of the invention the bit budget for segments of a first line 1 of an image to be compressed is fixed and equal for the segments 1_k of the first line, while the bit budget of segments $2_k, 3_k$ of further lines 2, 3,... is dependent on the complexity of the

image structure in a previous line, for example dependent on the complexity of an image structure in the adjacent line.

As already mentioned, pixel values of first pixels of first segments in each line are coded directly, instead of coding a difference. According to an example of the invention the positions of so-called "bad pixels" in one segment of a block are identified using differences calculated for adjacent pixels of a previous segment. These bad pixels will then be treated like the first pixels in the first segments of each line, i.e. their pixel values are coded directly. The position of bad pixels may be located by comparing the difference between two adjacent pixel values with further differences within the same segment. If this difference is significantly higher than the other differences or the mean value of the other differences, presence of a bad pixel is assumed. Under the assumption that the pixels of adjacent segments of one block are correlated, the position of a bad pixel in one segment corresponds to the position of a bad pixel in the adjacent segment.

For the above discussion it had been assumed that one pixel value is assigned to each pixel of a segment. In fact two pixel values, a luminance value (Y) and a first (U) or a second (V) chrominance value are assigned to each pixel. For the YUV422 format a bit stream to be compressed for each segment is $Y_1U_1 Y_2V_1 Y_3U_2 \dots$, whereas Y_k denote luminance pixel values and U_k, V_k denote chrominance pixel values. A segment may comprise 16 pixels, which results to 32 pixel values to be coded (compressed).

25

Assuming a YUV format, three types of differences are calculated: luminance difference DY, which is the difference of luminance values Y of two adjacent pixels, a first chrominance difference DU which is the difference of first chrominance values U of two pixels, and a second chrominance difference DV, which is the difference of second chrominance values V of two pixels. These differences are calculated as follows:

30

$$DY_n = Y_n - Y_{n-1} \quad (11a)$$

$$DU_n = U_n - U_{n-1} \quad (11b)$$

$$DV_n = V_n - V_{n-1} \quad (11c)$$

The method steps for a linewise compression of pixel values as
5 discussed above are applied to the three differences independently.
Consequently, there are three first pixel values Y_0 , U_0 , V_0 at the beginning of a
first segment in each line, that are coded directly.

10 Within one segment the same coding method is applied to pixel values of
one type, while pixel values of different types, i.e. luminance values on one
hand side and chrominance values on the other hand side, may be coded
differently.

As already explained, test encoding using a number of different coding
15 methods is performed prior to finally encoding an segment using one of these
methods. According to an example at least four different coding methods may
be applied. These different methods ,for example may include one method of
directly encoding pixel values by truncating LSBs and may include three
methods of VLC encoding differences of pixel values, where the pixel values
20 may be truncated or non.truncated pixel values. The direct coding method may,
for example, include truncating 4 LSBs, while the difference coding methods
may include truncation lengths of 0, 2 or 4 LSBs, where truncation length
denotes the number of bits being truncated from the pixel values prior to
calculating the differences. The coding method with 0 LSB truncation is
25 lossless. For the other difference coding methods coding the differences is
lossless, but the differences are lossy due to truncation of LSBs from the pixel
values prior to calculating the differences. The direct method in this example
has the worst quality, while the difference method with 0 LSB truncation has the
best quality. The quality of the difference methods decreases with the number
30 of truncated LSBs increasing. It goes without saying that additional direct or
difference encoding method, having higher or lower truncation lengths may be
considered for test encoding and finally encoding the segments.

The method according to the invention, for example, may be used for temporarily storing images during de-interlacing or motion compensated interpolation methods. In de-interlacing as well as in motion compensated de-interlacing information from a number of images of a image sequence are
5 required for image processing. In order to reduce memory space and in order to reduce the memory bandwidth required for storing the images in a memory device the images may be compressed prior to storing.

For storing the compressed images the compressed segments are
10 written into a memory buffer. Since the bit budget assigned to the individual segments may vary, the bit rate of a bit stream written into the memory buffer may either vary. According to a further embodiment of the invention the bit rate, and therefore the bit budget assigned to segments of one line may also vary dependent on the current load of the memory buffer. This helps avoiding the
15 buffer to run empty and avoiding the buffer to overflow.

Summarizing the above, embodiments of the invention relate to:

A display processing method for manipulating at least one received raw image to be displayed on a display device, comprising:

- 5 a method of scaling image for future display in a scheduled time slot;
 a method of de-interlacing of re-aligning pixels of the current image by referring to neighboring picture;

- an image encoder for compressing the 1st raw image to generate an associated compressed image to be stored in at least one buffer coupled to the
10 display control device; and

 at least one manipulation for generating an output image adapted to be displayed on the display device according to a decompressed image from the frame codec device by decompressing the compressed image.

- 15 A method for compressing groups of pixels, comprising:
calculating the difference between neighboring pixels segment by segment and store it into a temporary buffer;
marking the degree of complexity of each segment of one of the neighboring line of pixels; and
20 coding the differential values of adjacent pixels segment by segment by assigning bit rate according to the complexity of one of the neighboring lines.

- A method of compressing groups of pixels, comprising
calculating the difference between neighboring pixels segment by segment;
25 recording pixels with longer than original pixel length by using one bit for each color component and not compressing these pixels in the same position of the next line pixels; and
 compressing the differential values of the rest of adjacent pixels segment by segment by assigning bit number according to the complexity of one of the
30 neighboring lines.

A method of compressing groups of pixels and controlling output bit number of the compressed pixels for avoiding buffer overflow and underflow, comprising

- 5 applying a compression engine to reduce data rate of a group of pixels;
- deploying an anti-peak image buffer with a predetermined width and depth for temporarily storing the compressed pixel information with input from the compression engine and another node of output to other device;
- deploying an output bit rate counter to estimate the output bit number and the output flow rate of the output buffer; and
- 10 adjusting the compression rate segment by segment according to the information recorded in the bit rate buffer counter.

15 The image compression according to the invention may be used for compressing image data before storing the image data into a frame buffer which significantly reduces requirement of the density, bandwidth and power consumption of storage device.

- 20 The present invention further applies to the following concept of taking advantage of the correlation between neighboring lines of pixels which have higher similarity and less differential value to achieve low bit rate of storing the frame of pixels:
- Marking the pixels with large variance in a upper line and no compression in the current line with corresponding location, compressing others pixels referring
 - 25 to upper line complexity and assign bit rate of each segment accordingly;
 - Prediction and selection of the mode of separating MSB bits and LSB bits and no separation for achieving a shortest code for a VLC coding;
 - Coding the predicted differential value by a predicted divider without adding extra code and minimized code length of the Quotient and Remainder;
 - 30 - Applying a buffer control mechanism to avoid the output buffer from overflow or underflow;

The image compression compresses the on-chip image line buffer of display device controller which lines are used for scaling and de-interlacing.

5 The image compression includes method and apparatus of compressing the image data by a procedure of separating the MSB bits and LSB bits of the DPCM coded pixels difference between adjacent pixels.

10 According to an embodiment of the image buffer compression, the input pixels are tested to determine whether the MSB and LSB bits are compressed separately to gain higher compression rate.

15 According to an embodiment of the image buffer compression, the input pixels are tested to determine which VLC coding mode reaches the lowest bit rate and assigns the VLC coder to select that mode.

The image compression detects the variance of MSB bits of the DPCM code and determines whether the MSB and LSB bits to be compressed separately or not and which mode (8/0, 7/1, 6/2, 5/3 and 4/4).

20 According to an embodiment of the image compression, the distribution of recovering the truncated bits will not start from NOT the same position hence avoid accumulation of error over time.

25 According to an embodiment of the image compression, the block size of pixels are tradeoffs among compression rate, image quality, latency of accessing and ease of design.

30 According to an embodiment of the image compression, under the requirement of a fixed compression rate, the bit rate of Y (Luma) and U/V (Chroma) are put together as a compression unit in allocating the bit rate distribution and truncation.

According to an embodiment, a counter is applied to calculate and trace the compression rate of each line of a frame of pixels to ensure the fixed compression rate. When the accumulated compression rate of a certain of lines reached a value behind a preset target, higher compression rate will be applied to the next line or a couple of lines to gradually pull back the compression rate.

The video bit stream decoding analyzes the complexity and quality of the decompressed block pixels and decides which compression mode to be applied to reduce the data rate.

According to one embodiment the differential values of adjacent pixels will be adjusted to be all positive which reduces the bit requirement from 9 bits down to 8 bits.

According to another embodiment each color component (Y, U and V) is compressed and decompressed separately with the final data rate combined together to fit a predetermined bit rate.

According to another embodiment, each color component (Y, U and V) within a line has one bit as the mark "bad pixel: high complexity" and to represent action of "No compression".

According to another embodiment, each line pixels are compressed with variable length of bits segment to segment.

According to another embodiment the marker bits of the upper line will be downloaded to a smaller buffer segment-by-segment as a reference in compression and decompression.

According to another embodiment, a fixed clock cycle time will be assigned for continuously encoding and decoding a whole line of pixels.

According to another embodiment one encoder/decoder is assigned to encode and decode the Y component and another encoder/decoders will encode and decode U and V components in different cycle time.

5 According to another embodiment the adjusted differential value, D_n of adjacent pixels will be coded by dividing the D_n by a predicted divider to achieve a result of Quotient (Q) and a Remainder (R).

10 According to another embodiment the Quotient and Remainder are separated by inserting one "1" and the value of Q and R turn out to be number of "0s" (Example $17 = 3 \times 5 + 2$, $Q=3$, $R=2$, the code will be "000100")

According to another embodiment the divider is predicted without inserting bits into represent which reduces the bit rate.

15 According to an embodiment of the present invention, a predetermined number of dividers are applied and tested to determine which reaches the lowest data rate which will be selected as the divider for coding the predicted differential values of pixels.

20 According to one embodiment a code (Ex. "00000000" is inserted to represent that the Remainder of its following 8 bits is the same value of the predicted D_n to be coded which limited the max. length of code to be 16 bits.

25 According to another embodiment multiple calculation engines are implemented to pipelining predict the mode of VLC coding and continuously compressing the adjusted differential values of neighboring pixels.

According to an embodiment of efficient lossless data stream coding, the initial divider number is calculated by a statistical number of previous samples.

30

According to an embodiment the divider of the next pixel and the Quotient and Remainder of the current pixels are calculated in the same cycle time.

According to another embodiment one single segment pixel buffer is implemented to receive input pixels and providing pixels to the compression engine pipelining without stopping and waiting.

CLAIMS

1. A method for compressing image data of an image having lines of pixels, comprising:
 - 5 subdividing the lines into segments, with each of the segments comprising a number of pixels, with each pixel being assigned at least one pixel value,
 - assigning a bit budget to each segment,
 - compressing the segments for obtaining compressed segments using
 - 10 one of at least two coding methods having different coding qualities, with the coding method being selected such that a bit number of the compressed segment is equal to or lower than the bit budget,
 - wherein the bit budget of a segment of one line is selected dependent on a coding quality obtained for an adjacent segment in an adjacent line.
 - 15
2. The method of claim 1, wherein the method further comprises:
 - detecting the worst segments in one line being compressed with a worst coding quality as compared to further segments of the line, and detecting best
 - 20 segments in one line being compressed with a best coding quality as compared to further segments of the line,
 - in an adjacent line increasing the bit budget for at least some segments being adjacent to worst blocks as compared to the bit budget of the worst blocks, and decreasing the bit budget for at least some segments being adjacent to best blocks as compared to the bit budget of the worst blocks.
 - 25
3. The method of claim 2, wherein the number of segments for which the bit budget is increased from line to line equals the number of blocks for which the bit budget is decreased.
- 30 4. The method of one of the preceding claims, wherein a fixed bit budget is assigned to each block of a first line of a given image.

5. The method according to one of the preceding wherein the at least two coding methods include at least one variable length coding method.
6. The method according to claim 5, wherein the at least one variable
5 length coding method for a given pixel comprises:
calculating the difference between the given pixel and a prediction value;
dividing the difference by a divisor (M) to obtain an integer (Q) and a remainder (R);
mapping the integer and the remainder to a code.
10
7. The method according to claim 6, wherein the prediction value is the pixel value of an adjacent pixel.
8. The method according to claim 6 or 7, wherein the divisor (M) is updated
15 on a bit by bit basis within each segment, starting from an initial value.
9. The method according to claim 8, wherein an updated divisor (M) is dependent on a current divisor and a current pixel value.
- 20 10. The method according to one of claims 5 to 9, wherein prior to the difference at least one least significant bit is truncated from binary words representing the pixel value and the prediction value.
11. The method according to one of claims 5 to 10, wherein prior to variable
25 length encoding the differences are mapped to target values representing the differences.
12. The method according to one of the preceding claims, wherein the at least one coding method includes at least one fixed length coding method.
30
13. The method according to claim 12, wherein the at least one fixed length coding method includes truncating a given number of least significant bits from

a binary word representing a pixel value and using the truncated binary word as a code word representing the pixel value.

14. The method according to any of the preceding claims, wherein bad pixels
5 are selected, with the pixel values of bad pixels being coded directly.

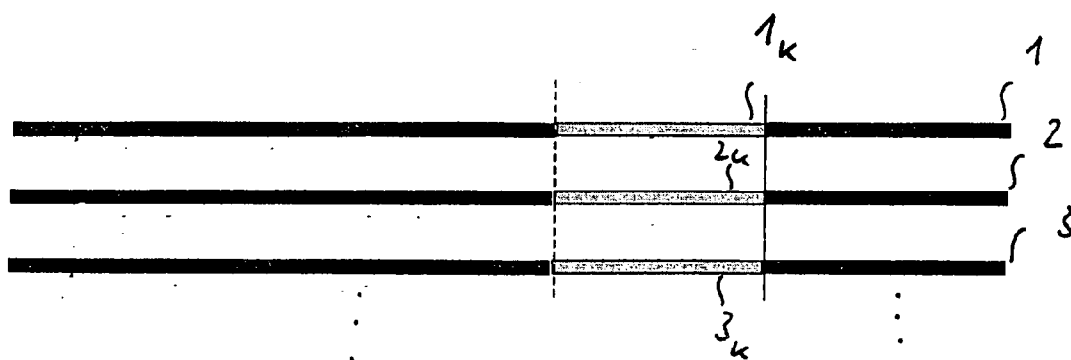
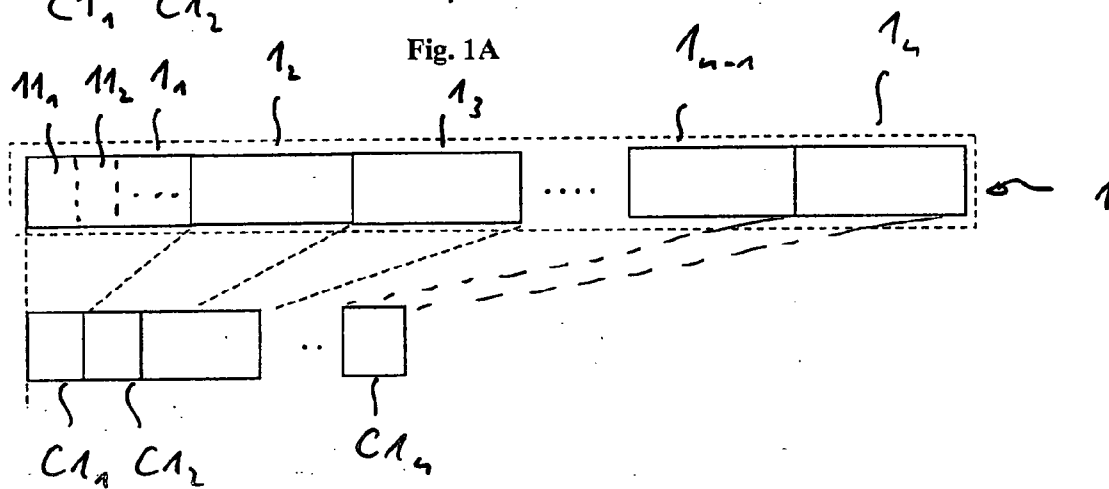
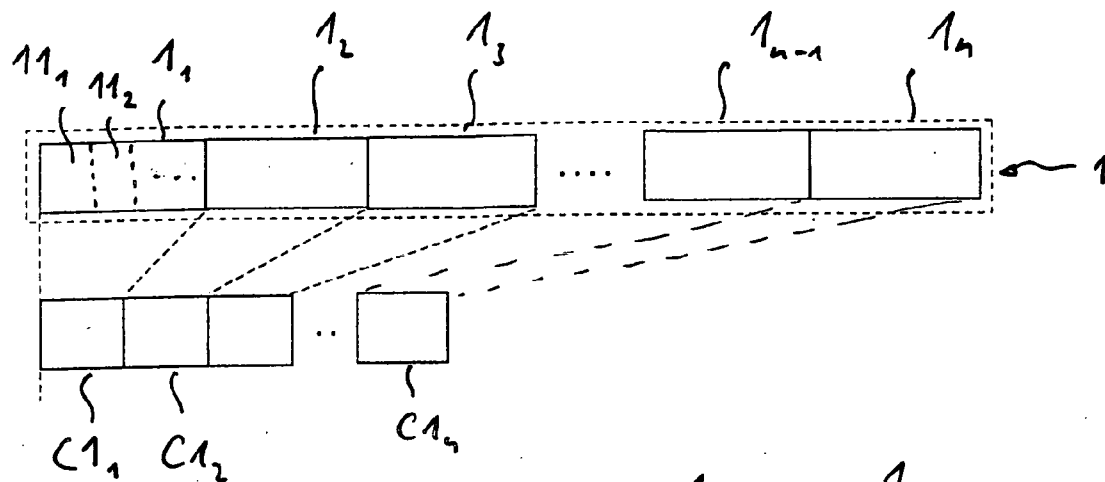
15. The method of claim 14, wherein selecting a bad pixel in a given segment comprises the steps of:

10 evaluating the difference between a pixel value of a given pixel and a prediction value in a segment adjacent to the given segment,
comparing the difference to other differences between pixel values and prediction values of other pixels in the adjacent segment,
marking a pixel in the given segment having the same position as the given pixel in the adjacent segment to be a bad pixel dependent on the
15 comparison result.

16. The method according to one of the preceding claims, comprising:
prior to coding pixel values or differences of a given segment performing
a test encoding using the at least two coding methods, in order to select the
20 coding method, which, given the bit budget for the segment, results to the best coding quality,
applying the selected coding method for coding pixel values of the block.

17. An apparatus for compressing image data of an image having lines of
25 pixels, comprising:
means for subdividing the lines into segments, with each of the segments comprising a number of pixels, with each pixel being assigned at least one pixel value,
means for assigning a bit budget to each segment,
30 means for compressing the segments for obtaining compressed segments using one of at least two coding methods, with the coding method being selected such that a bit number of the compressed segment is equal to or lower than the bit budget,

means for selecting the bit budget of a segment of one line dependent on a coding quality obtained for an adjacent segment in an adjacent line.



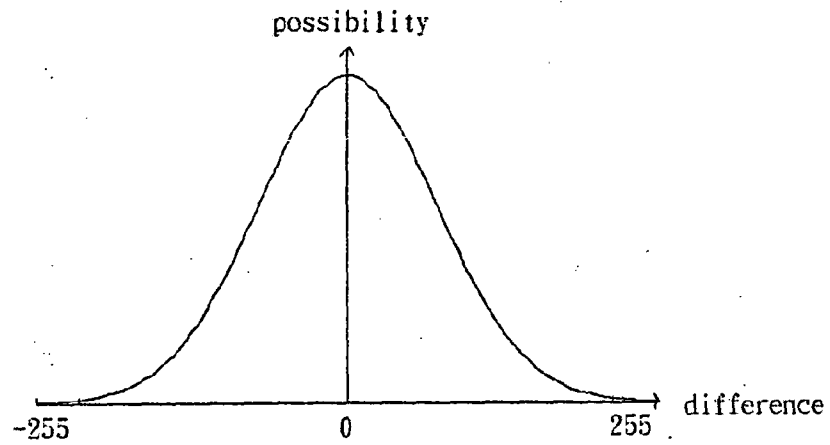


FIG 3

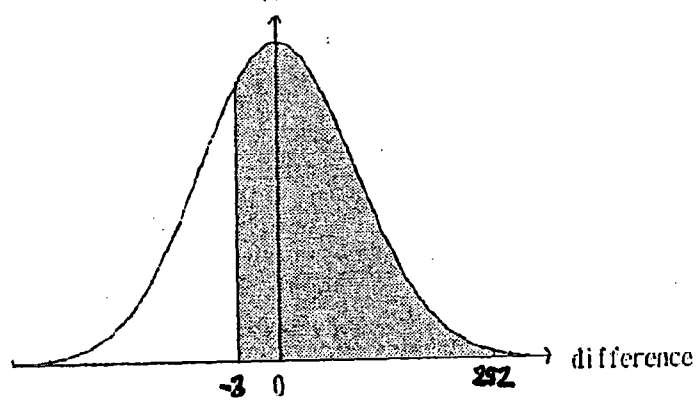


FIG 4A

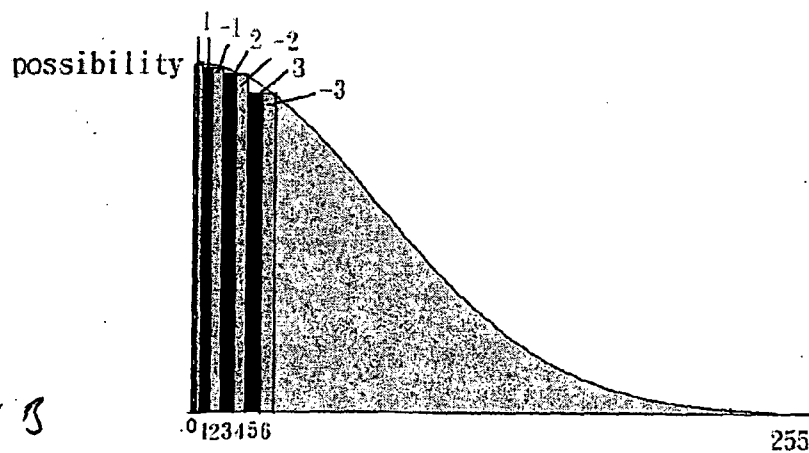


FIG 4B

Code	00....0  Q times	1	R (binary code, x bits)
------	---	---	----------------------------

FIG 5

D	m	code		
		Q	Marker	R
27	8	000	1	011
6	16	-	1	0110
0	1	-	1	-
10	1	0000000000	1	-

FIG 6

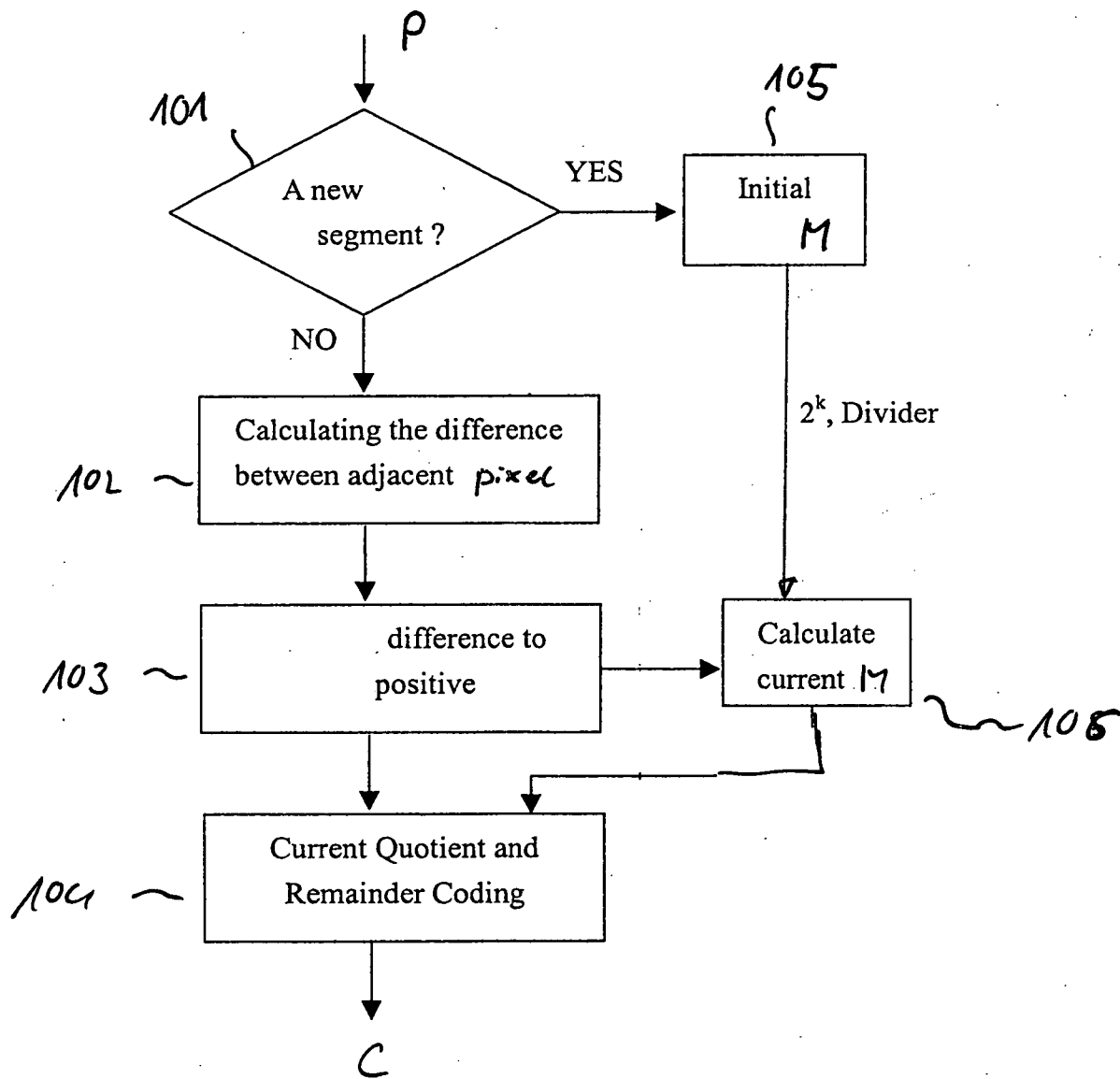


FIG 7

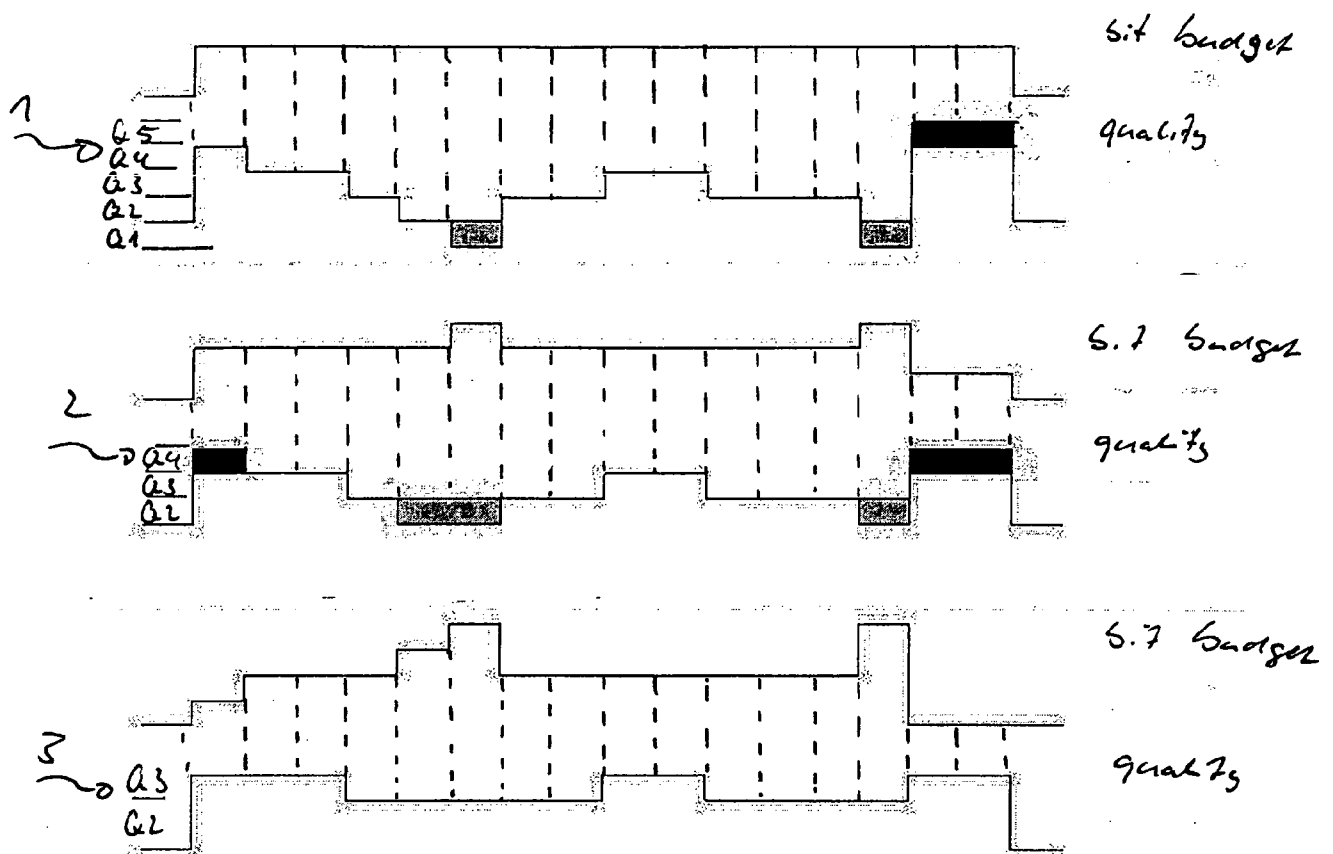


FIG 8

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2007/002158

A. CLASSIFICATION OF SUBJECT MATTER

INV. G06T9/00 H04N7/26 H04N7/30 H04N1/417

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06T H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	EP 0 124 926 A1 (PHILIPS NV [NL]) 14 November 1984 (1984-11-14) abstract page 1, line 1 - page 2, line 18 page 8, line 31 - page 10, line 22; claim 1	1-17
Y	US 6 088 392 A (ROSENBERG JONATHAN DAVID [US]) 11 July 2000 (2000-07-11) abstract column 2, line 10 - column 2, line 44 column 6, line 21 - column 8, line 46 claim 1	1-17
A	US 6 778 291 B1 (CLOUTHIER SCOTT C [US]) 17 August 2004 (2004-08-17) abstract	1-17
	----- -/--	

☒ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

25 May 2007

Date of mailing of the international search report

05/06/2007

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Schoeyer, Marnix

INTERNATIONAL SEARCH REPORT

International application No
PCT/EP2007/002158

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2004/091160 A1 (HOOK TIMOTHY VAN [US] ET AL) 13 May 2004 (2004-05-13) abstract -----	1-17

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/EP2007/002158

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0124926	A1	14-11-1984	CA 1254999 A1 30-05-1989
			DE 3461879 D1 05-02-1987
			JP 5076171 U 15-10-1993
			JP 59200581 A 13-11-1984
			NL 8301264 A 01-11-1984
			US 4622585 A 11-11-1986
US 6088392	A	11-07-2000	NONE
US 6778291	B1	17-08-2004	NONE
US 2004091160	A1	13-05-2004	NONE