(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
22 December 2005 (22.12.2005)

PCT

(10) International Publication Number
WO 2005/122587 A1

(51) International Patent Classification[7]: H04N 7/26, 7/36

(21) International Application Number:
PCT/IB2005/051845

(22) International Filing Date: 7 June 2005 (07.06.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
04300323.5 11 June 2004 (11.06.2004) EP

(71) Applicant *(for all designated States except US)*: **KONINKLIJKE PHILIPS ELECTRONICS N.V.** [NL/NL]; Groenewoudseweg 1, NL-5621 BA EINDHOVEN (NL).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **BOURGE, Arnaud** [FR/FR]; Société Civile SPID, 156 Bd Haussmann, F-75008 Paris (FR). **JUNG, Joël** [FR/FR]; Société Civile SPID, 156 Bd Haussmann, F-75008 Paris (FR).

(74) Agent: **CHAFFRAIX, Jean**; Société Civile SPID, 156 Boulevard Haussmann, F-75008 Paris (FR).

(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, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

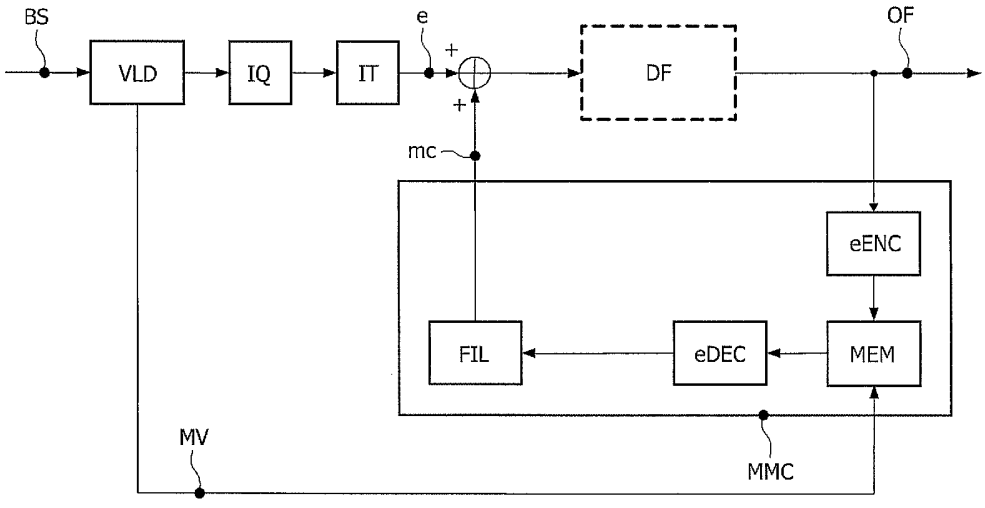
(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, 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).

Declaration under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ,*

[Continued on next page]

(54) Title: METHOD OF STORING PICTURES IN A MEMORY USING COMPRESSION CODING AND COST FUNCTION INCLUDING POWER CONSUMPTION



(57) Abstract: The present invention relates to a method of storing pictures in a memory, a picture being divided into data blocks, said method comprising the step of: computing a transform of an input data block for producing a transformed data block comprising a set of n transformed elements, where n is an integer, entropy coding m first transformed elements of the transformed data block for producing an encoded data block, where m is an integer lower than n, computing a cost function on the basis of a weighted sum of a distortion value between the input data block and the encoded data block and a power consumption required for reading or writing the encoded data block in the memory, iterating the entropy coding and cost function computing steps for different values of m, and storing in the memory the encoded data block corresponding to the value of m that minimizes the cost function.

WO 2005/122587 A1

CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW, ARIPO patent (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI patent (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.

METHOD OF STORING PICTURES IN A MEMORY USING COMPRESSION CODING AND COST FUNCTION INCLUDING POWER CONSUMPTION

FIELD OF THE INVENTION

The present invention relates to a method of and a device for storing pictures in a memory, said pictures being processed according to a predictive block-based technique.

This invention may be used, for example, in video decoders, video encoders or portable apparatuses, such as personal digital assistants or mobile phones, said apparatuses being adapted to decode or to encode pictures.

BACKGROUND OF THE INVENTION

Low power consumption is a key-feature of mobile devices. Mobile devices now provide video encoding and decoding capabilities that are known to dissipate a lot of energy. So-called low-power video algorithms are thus needed.

As a matter of fact, accesses to an external memory such as SDRAM are a bottleneck for video devices. This is due both to power consumption issues, as memories are known to be the most power-consuming part of a system, and to speed limitation because of the bandwidth available for the exchanges between a central processing unit CPU and the memory.

In conventional video decoders, the motion compensation unit needs many such accesses because it constantly points to blocks of pixels in so-called reference frames. To overcome this problem, so-called "embedded compression" has been proposed. Said embedded compression has originally been developed to decrease the memory size at the expense of a quality decrease, due to lossy compression of the reference frames.

An example of embedded compression is shown in Figure 1 applied to an H.264 video decoder. Said video decoder comprises in series:

- a variable length decoding block VLD suitable for decoding an encoded bit-stream BS so as to produce decoded data blocks on the one hand, and decoded motion vectors MV on the other hand,
- an inverse quantizing block IQ suitable for producing quantized data blocks,
- an inverse frequency transform block IT, for example in inverse discrete cosine transform block IDCT, for producing inversely transformed data blocks corresponding to a residual error data block e.

The video decoder further includes an adder for adding a motion-compensated data block to a residual error data-block. The motion-compensated data blocks are produced by a modified motion compensation unit MMC comprising in series an embedded encoding unit eENC for producing encoded data blocks, an image memory MEM for storing said encoded data blocks, an embedded decoding unit eDEC and an interpolation filter FIL. The output of the adder is a decoded data block of the output decoded image OF which is then delivered to a display (not represented) and which is also delivered to the embedded encoding unit eENC.

It has been proven in "A Low-Power H.264 Decoder with Graceful Degradation", by A. Bourge and J. Jung, Proceedings Of Electronic Imaging, VCIP, January 2004, that, given some specific requirements, embedded compression techniques help reducing memory transfers and hence power dissipation. One requirement is that the compression ratio maps the memory structure. This requirement means that a compressed block is stored at convenient access points in the memory. For instance the start address of each encoded data block is word-aligned for a SDRAM memory comprising conventionally words having a predetermined length (e.g. 16, 32 or 64 bits), and the size of the encoded data block is optimized if it fits the size of data requests, i.e. bursts of 1, 2, 4, 6 or 8 words are extracted from the memory during one reading cycle.

A conventional way to respect the above requirement could be to set a fixed compression ratio for each block in order to fit in a data burst. For instance, the reference frame is split into data blocks of 8x8 luminance Y and 4x4 chrominance U and V pixels, said luminance and chrominance components being sampled on 8 bits. Therefore, a data block corresponds to 768 uncompressed bits. If we set a compression factor of 3, one data block gets encoded based on a bit budget allocation of 256 bits, which exactly matches the structure of a memory platform with a 32-bit data bus that accepts 8-word bursts.

However, such a fixed compression ratio method is not optimal in terms of visual quality. Indeed, too few bits might be used to correctly encode complex data blocks resulting in a great loss of information, while some less complex data blocks can be encoded without loss with less than 256 bits resulting in a waste of bits.

SUMMARY OF THE INVENTION

It is an object of the invention to propose a method of and device for storing pictures in a memory which allows a better visual quality than the one of the prior art to be achieved while keeping a reasonable power consumption of the memory.

To this end, the method in accordance with the invention is characterized in that it comprises the steps of:

- computing a transform of an input data block for producing a transformed data block comprising a set of n transformed elements, where n is an integer,
- entropy coding m first transformed elements of the transformed data block for producing an encoded data block, where m is an integer lower than n,
- computing a cost function on the basis of a weighted sum of a distortion value between the input data block and the encoded data block and a power consumption required for reading or writing the encoded data block in the memory,
- iterating the entropy coding and cost function computing steps for different values of m, and
- storing in the memory the encoded data block corresponding to the value of m that minimizes the cost function.

The introduction of a cost function based on power-rate-distortion criterion (i.e. a distortion criterion which is dependent on distortion and power consumption) allows a better tradeoff between power consumption, memory bandwidth and visual quality.

Beneficially, the cost function is computed on the basis of a weighted sum of the distortion value, the power consumption and a number of bits of the encoded data block.

The present invention also relates to a storage device implementing such a storage method.

According to an embodiment of the invention, the input data block is transformed and entropy encoded in a first pass, and the resulting encoded data block is truncated at a bit position corresponding to the value of m for which the cost function is minimum, and then stored in the memory.

According to another embodiment of the invention, a variation of the cost function is computed each time a new transformed element is to be added to a current encoded data block, the current encoded data block being stored in the memory as soon as the variation is positive.

According to another embodiment of the invention, the memory is accessed by bursts of i words of predetermined length, where i is a variable integer, and the power consumption is derived from the number of bits of the encoded data block, the power consumption per bit in burst of i words, the value of i, the number of bursts of i words and the length of words. In this case, the cost function is then particularly adapted to the structure and properties of the memory and data bus.

The present invention also relates to a video decoder for decoding a bit-stream, said decoder comprising a decoding unit for providing error data blocks, said storage device for storing encoded data blocks, means for extracting at least one encoded data block from the memory and for decoding said at least one encoded data block so as to deliver a current motion-compensated data block, and an adder for adding a current error data block to the current motion-compensated data block, the output of said adder being provided to the input of the storage device.

The present invention also relates to a video encoder for encoding a sequence of pictures, a picture being divided into input data blocks, said encoder comprising an encoding unit for providing a partially encoded data block, a decoding unit for providing a partially decoded data block from the partially encoded data block, a prediction unit comprising in series a storage device and means for extracting at least one encoded data block from the memory and for decoding said at least one encoded data block so as to deliver a motion-compensated data block, an adder for adding the motion-compensated data block to the partially decoded data block, the output of said adder being provided to the input of the prediction unit, and a subtracter for subtracting the motion-compensated data block from an input data block, the output of said subtracter being provided to the input of the encoding unit.

The invention also relates to a portable apparatus comprising such a storage device.

Said invention finally relates to a computer program product comprising program instructions for implementing said storage method.

These and other aspects of the invention will be apparent from and will be elucidated with reference to the embodiments described hereinafter.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will now be described in more detail, by way of example, with reference to the accompanying drawings, wherein:

- Figure 1 shows a block diagram of an embodiment of a decoding device; and
- Figure 2 shows a block diagram of an embodiment of an encoding device.

DETAILED DESCRIPTION OF THE INVENTION

The present invention introduces a new way of dealing with power consumption constraints for embedded compression schemes in video decoders and encoders. Embedded compression consists in compressing the reference frames in a predictive video coding

scheme, as described before. The invention proposes to perform a bit budget allocation of the embedded compression scheme based on a power-rate-distortion criterion.

The present invention can be applied to any video encoding or decoding device where sequences have to be stored in memory. Indeed, a decoded frame generally needs to be stored in the memory so that it can be later retrieved to predict the next frame(s) through motion compensation. The invention is particularly interesting for reducing the size of the reference image memory while keeping a sufficient overall image quality of the output decoded image. For sake of clarity, we will focus in the following description on the case of a conventional video decoder (for example MPEG-2, MPEG-4, H.264, or the like).

Let us consider again the video decoder of Figure 1. Said video decoder is adapted to decode a bit-stream comprising a sequence of encoded pictures, said encoded pictures being divided into encoded data blocks. For that purpose, said decoder comprises:

- a decoding unit comprising in series a variable length decoding block VLD, an inverse quantization block IQ and an inverse transform block IT (e.g. adapted to perform an inverse discrete cosine transform IDCT), said decoding unit being suitable for providing error data blocks e from the bit-stream and decoded motion vectors MV,
- a motion compensation unit MMC for performing motion compensation based on decoded data blocks and motion vectors so as to deliver motion-compensated data blocks, and
- an adder for adding a current decoded error data block to a current motion-compensated data block so as to deliver a current decoded data block.

The decoded output image OF corresponding to a group of decoded data blocks is then delivered to a display (not represented). The decoding device optionally comprises a de-blocking filter DF, said filter being for example the one proposed in the H.264 standard.

The reconstruction unit of the video decoder in accordance with the invention comprises in series an embedded encoding unit eENC, a memory MEM, an embedded decoding unit eDEC and an interpolation filter FIL.

According to an embodiment of the invention, the embedded encoding eENC unit comprises a transform block for frequency transforming (e.g. using a discrete cosine transform DCT) an input data block so as to produce a transformed data block comprising a set of 64 transformed coefficients (e.g. DCT coefficients) in the case of luminance blocks of 8x8 pixels, each coefficient being sampled on 8 bits.

It further comprises an entropy coding block, for entropy coding m first transformed elements of the transformed data block so as to produce an encoded data block, where m is an integer varying from 1 to n and where n is the total number of relevant transformed elements in the transformed data block. In said embodiment, the entropy-coding block is based on bit plane coding. Said coding encodes DCT coefficients bit plane by bit plane, starting with the Most Significant Bits. Such a coding scheme is for example described in "Low-complexity lossless and fine-granularity scalable near-lossless compression of color images", by R.J. van der Vleuten, Proceedings of the Data Compression Conference, pp.477, April 2002. According to such a coding scheme, the DC coefficients are losslessly encoded. For the AC coefficients, compression is achieved through a zonal coding: for each bit plane the significance map of the previously insignificant coefficients is represented by the maximum row Rmax and the maximum column Cmax where a bit is equal to 1. The data beyond this zone does not need to be transmitted. Since the energy usually lies in the low frequency coefficients, this technique implies a good bit-rate reduction. Its great advantage is to provide a fine-grain scalability. With said embedded encoding scheme, each transformed element is a bit of a DCT coefficient and n is equal at most to 512 bits.

The video encoder comprises computing means (not represented), e.g. a processor, for computing a cost function on the basis of a weighted sum of a distortion value between the input data block and the corresponding encoded then decoded data block, a number of bits of the encoded data block, and/or a power consumption required for reading or writing the encoded data block in the memory.

The cost function c(i+1) is computed as follows:

$c(i+1) = \lambda 1 * size\_in\_bits + \lambda 2 * distortion + \lambda 3 * power\_consumption$

$c(i+1) = c(i) + \delta c$

where:

- size_in_bits is the number of bits of a next encoded data block if a next transformed element a(i+1) is added to the current encoded data block;
- distortion is a computation of a difference (e.g. mean square errors MSE, sum of absolute differences SAD or the like) between the input data block and the corresponding decoded version of the encoded data block; said distortion can also be computed without decoding the encoded block, for example in the case of a convention encoding chain DCT-Q-VLC, said distortion is derived from a difference, e.g. MSE, between the DCT coefficients before and after quantization, which is

equivalent to the MSE between the input data block and the decoded version of the encoded data block;

- power_consumption is an estimation or computation of the power dissipation for writing the encoded data block in the memory and for reading it during a further motion compensation step;
- c(i+1) represents the value of the cost function when a set of (i+1) transformed elements of a next data block have been encoded, c(i) represents the value of the cost function when a set of i transformed elements of the current data block have been encoded, and δc is the cost variation between these two values;
- $\lambda 1$, $\lambda 2$, and $\lambda 3$ are weighting factors that reflect the relative importance of each parameter.

According to an embodiment of the invention, the next transformed element a(i+1) is actually added to the bit-stream if and only if the value of δc is negative. It both depends on properties of the current data block through the parameters size_in_bits, distortion and power_consumption, and on the encoding strategy through $\lambda 1$, $\lambda 2$, and $\lambda 3$. For instance, a strong value of $\lambda 1$ is needed when there is a strict requirement on the final size of the compressed frame, i.e. it should not exceed a predetermined size, for example the total size of the SDRAM. A large value of $\lambda 2$ means that visual quality is the main decision criterion. Finally, if the value of $\lambda 3$ is high, it means that a gain in power dissipation is more important that a loss of information. There are several advantages in using such an approach.

A first advantage is the visual quality. The use of embedded compression in a video decoder makes the sequence of pictures shift away from the standard decoding (the so-called drift effect), implying a quality degradation that accumulates through the group of pictures. Compared to a fixed compression ratio solution, as described in the prior art, the proposed method allows a more flexible rate allocation, giving more bits to more complex blocks, as in a conventional rate-distortion approach.

Another advantage is the power consumption and the memory structure. Compared to a conventional rate-distortion criterion, the use of the proposed cost function takes into account the power dissipation induced by storing more bits. In practice, this dissipation is calculated (or measured) thanks to known properties of the memory implemented in the device, such as its structure (i.e. the word size), the width of the data bus, the available burst modes, and the energy consumption of an access in each burst mode (read or write). Hence, the bit-rate allocation process cannot favor a small rate-distortion gain that would imply a large energy loss.

Still another advantage is the power scalability. If needed, the weighting factor λ3 is automatically tuned according to the level of the device battery. At full charge, the value of λ3 is small and the cost function is very close to a conventional rate-distortion criterion, optimizing visual quality. At near-empty charge, power saving is the critical issue and thus the value of λ3 is set at a maximum value.

The video decoder comprises a memory controller (not represented) which enables to write in the memory the encoded data block corresponding to the value of m that minimizes the cost function. According to the embodiment described above, the variation of the cost function is computed each time a new element is to be added to the bit-stream. The process stops as soon as the variation is positive. This method is less computational expensive but the process can be trapped in a local minimum of the cost function. According to another embodiment of the invention, the input data block is fully encoded in a first pass, and the resulting bit-stream is locally stored at a temporary location in the memory. For each value of m, the cost function is computed. The bit-stream is truncated at a bit position corresponding to the m first elements for which the cost function is minimum, and then written in the memory. On the one hand this method is exhaustive and thus ensures to find the truncation point with the overall minimum cost. On the other hand it is computational expensive because all the information has to be coded until the last bit and all the values of the cost function have to be computed and stored.

Based on a current decoded motion vector MV, the memory controller enables to read at least one encoded data block from the memory. Then, the embedded decoding unit eDEC is adapted to decode the at least one encoded data block. Depending on the embedded encoding unit, the embedded decoding unit comprises an entropy-decoding block and an inverse transform block IT blocks in series. The at least one decoded data block is finally processed by an interpolation filter FIL so as to deliver a current motion-compensated data block mc, according to a principle known to a person skilled in the art.

According to another embodiment of the invention, the embedded encoding unit eENC comprises a conventional encoding chain including in series a transform block (e.g. using DCT), a quantization block and an entropy-coding block (e.g. a variable length decoding VLC block). In this case, the encoded data block comprises m first transformed elements which are the first m DCT coefficients, and the encoded data block corresponding to the value of m that minimizes the cost function is stored in the memory. The corresponding embedded decoding unit then comprises a conventional decoding chain

including in series a variable length decoding VLD block, an inverse quantization block and an IDCT block.

It will be apparent to a person skilled in the art that other compression schemes are possible. For example, the embedded encoding unit may comprise a transform block based on Differential Pulse Code modulation DPCM in series with an entropy-coding block of the VLC type.

In this paragraph, the step of computing the cost function is depicted in more detail. The parameter size_in_bits is known at each step by construction. The distortion value is computed building the truncated DCT coefficients and calculating the difference with the original values (using e.g. an algorithm based on the mean square errors MSE or the sum of absolute differences SAD).

The power_consumption is computed as follows. In modern SDRAM, data can be accessed by bursts of i words (for example i = 1, 2, 4, 6 or 8). A word contains N bits (N=32 in general, 16 and 64 are other common values), N is also the width of the data bus. The encoded block is written and fetched using n_8 bursts of 8 words, n_6 bursts of 6 words and so on. Each burst mode correspond to a different energy consumption. Table 1 gives an example of consumption per access in the case of an SDRAM with a 32-bit data bus width, with a burst of 1 and 8 words. The presented figures are estimations using a specific model.

Consumption per access	Burst of 1 word	Burst of 8 words
Reading	19020.7 pj	63943.2 pj
Writing	17848.8 pj	54568.0 pj

Table 1: SDRAM consumption estimation per access.

One can remark that the consumption of a burst of 8 words is much lower than the consumption of 8 simple accesses. Hence the power cost per transferred bit is different in each mode. For a given number of bits (size_in_bits), we have thus to select a combination of bursts that permits to transfer all the bits with a minimal dissipation according to the following equations:

$$\begin{cases} \sum_{i=1,2,4,6,8} i*n_i*N \geq size\_in\_bits & (1) \\ P = \sum_{i=1,2,4,6,8} \alpha_i*i*n_i*N \quad is\ min imized & (2) \end{cases}$$

where α_i is the power cost per bit (or per word) in burst mode i, given by the memory properties.

According to an embodiment of the invention, the optimal $\{n_8, n_6, \ldots, n_1\}$ configuration is determined based on the following iterative method, that employs a top-bottom approach. Let us assume that α_i decreases with i, which is always the case in practice. At each iteration, we consider two vector candidates: $n^{(0)}= \{n_8^{(0)}, n_6^{(0)}, \ldots, n_1^{(0)}\}$ and $n^{(1)}= \{ n_8^{(1)}, n_6^{(1)}, \ldots, n_1^{(1)}\}$. The first two candidates are:

$n^{(0)}= \{$ size_in_bits/(8*word_length), 0, ..., 0$\}$ and

$n^{(1)}= \{$ 1+size_in_bits/(8*word_length), 0, ..., 0$\}$.

If $n^{(0)}$ exactly fits size_in_bits, then it is the best candidate and the process stops. Otherwise, $n^{(0)}$ is too small to fulfill the first condition of Equation (1), whereas $n^{(1)}$ is large enough. The corresponding power consumption $P^{(1)}$ is computed. $n^{(1)}$ and $P^{(1)}$ are stored as the best candidates n_{best} and P_{best}.

At an iteration s, $n^{(0)}$ and $n^{(1)}$ are built up from the previous $n^{(0)}$ vector at iteration (s-1). The (s-1) values $n_8^{(0)}, n_6^{(0)}, \ldots, n_i^{(0)}$ are frozen and definitely chosen. The two candidates have now the following structures:

$n^{(0)}= \{n_8^{(0)}, n_6^{(0)}, \ldots, n_i^{(0)}, n_{i-1}^{(0)}, 0, \ldots, 0\}$ and

$n^{(1)}= \{ n_8^{(1)}, n_6^{(1)}, \ldots, n_i^{(1)}, n_{i-1}^{(1)}, 0, \ldots, 0\}$,

where $n_{i-1}^{(0)}=\delta n/((i-1)*N)$, $n_{i-1}^{(1)}= n_{i-1}^{(0)}+1$ and δn is the number of bits that were missed by the previous $n^{(0)}$ vector at iteration (s-1). If $n^{(0)}$ exactly fits size_in_bits, then it is the best candidate and the process stops. Otherwise, $n^{(0)}$ is too small to fulfill the first condition of Equation (1), whereas $n^{(1)}$ is large enough, as in the first iteration. The corresponding power consumption $P^{(1)}$ is computed. If $P^{(1)}$ is lower than the current P_{best}, then $n^{(1)}$ and $P^{(1)}$ are stored as n_{best} and P_{best}. The process goes on until the last burst mode is reached or until $n^{(0)}$ exactly fits size_in_bits.

A simpler, though non-optimal, solution consists in directly building the burst configuration through a pre-established rule, e.g. decomposing the encoded data block according to the following principle: $n = \{n_8, n_6, ..., n_1\}$ with n_8 = size_in_bits/(8*N), $n_i = \delta n_i/(i*N)$ and δn_i is the remaining number of bits; and in calculating the associated power_consumption. This value is then injected in the cost function. Such a solution has the advantage of being very low-cost in complexity, but does not ensure that the selected configuration minimizes power consumption for the transfer (read/write) of a given encoded data block.

A trade-off between the two above solutions would consist in building in advance the optimal configurations for all the size_in_bits values, and store them in a table (for instance

in a ROM). Anyway, the first described method is not so complex if the number of available burst modes is limited, which is the case in practice.

In the previous description, it has been assumed that the quantization step has already been selected if any. The resulting bit-stream has been made by truncating it at a relevant point, as described before, for instance after a VLC-encoded Run-Level couple.

According to another embodiment or in addition to the previous description, the cost function can be used to a priori select the quantization step QP. Size_in_bits is estimated using QP and the activity of the current data block (based on an average value of MSE or SAD). Distortion is estimated from the same two parameters. Finally, power_consumption is estimated as described before using the estimated size_in_bits value. The QP value that minimizes the cost estimation is selected.

Figure 2 shows an example of a video encoding device. Such an encoding device comprises:

- an encoding unit comprising in series a direct frequency transform block, for example a direct discrete cosine transform DCT, and a quantizing block Q suitable for transforming input video data IN into partially encoded data blocks; and variable length coding block VLC suitable for producing a bit-stream ES from the partially encoded data blocks;
- a decoding unit comprising in series an inverse quantizing block IQ; an inverse frequency transform block IT, for example an inverse discrete cosine transform block IDCT, for providing partially decoded data blocks from the partially encoded data blocks,
- a prediction unit comprising in series an embedded encoding unit eENC, a memory MEM, an embedded decoding unit eDEC and a motion compensation unit MC for producing motion-compensated data blocks, encoded data blocks produced by the embedded encoding unit being stored in the memory according to the method in accordance with the invention,
- an adder for adding the motion-compensated data block to the partially decoded data block, the output of said adder being provided to the input of the prediction unit, and
- a subtracter for subtracting the motion-compensated data block from an input data block, the output of said subtracter being provided to the input of the encoding unit.

Several embodiments of the present invention have been described above by way of examples only, and it will be apparent to a person skilled in the art that modifications and variations can be made to the described embodiments without departing from the scope of the

invention as defined by the appended claims. Further, in the claims, any reference signs placed between parentheses shall not be construed as limiting the claim. The term "comprising" does not exclude the presence of elements or steps other than those listed in a claim. The terms "a" or " an" does not exclude a plurality. The invention can be implemented by means of hardware comprising several distinct elements, and by means of a suitably programmed computer. In a device claim enumerating several means, several of these means can be embodied by one and the same item of hardware. The mere fact that measures are recited in mutually different independent claims does not indicate that a combination of these measures cannot be used to advantage.

CLAIMS

1. A method of storing pictures in a memory, a picture being divided into data blocks, said method comprising the step of:

- computing a transform of an input data block for producing a transformed data block comprising a set of n transformed elements, where n is an integer,
- entropy coding m first transformed elements of the transformed data block for producing an encoded data block, where m is an integer lower than n,
- computing a cost function on the basis of a weighted sum of a distortion value between the input data block and the encoded data block and a power consumption required for reading or writing the encoded data block in the memory,
- iterating the entropy coding and cost function computing steps for different values of m, and
- storing in the memory the encoded data block corresponding to the value of m that minimizes the cost function.

2. A method as claimed in claim 1, wherein the cost function is computed on the basis of a weighted sum of the distortion value, the power consumption and a number of bits of the encoded data block.

3. A method as claimed in claim 1, wherein the input data block is transformed and entropy encoded in a first pass, and wherein the resulting encoded data block is truncated at a bit position corresponding to the value of m for which the cost function is minimum, and then stored in the memory.

4. A method as claimed in claim 1, wherein a variation of the cost function is computed each time a new transformed element is to be added to a current encoded data block, the current encoded data block being stored in the memory as soon as the variation is positive.

5. A method as claimed in claim 1, wherein the memory is accessed by bursts of i words of predetermined length, where i is a variable integer, and wherein the power consumption is derived from the number of bits of the encoded data block, the power

consumption per bit in burst of i words, the value of i, the number of bursts of i words and the length of words.

6. A storage device for storing pictures in a memory, a picture being divided into data blocks, said device comprising:

- an embedded encoding (eENC) unit for transforming an input data block so as to produce a transformed data block comprising a set of n transformed elements, where n is an integer, and for entropy coding m first transformed elements of the transformed data block so as to produce an encoded data block, where m is an integer lower than n,
- computing means for computing a cost function on the basis of a weighted sum of a distortion value between the input data block and the encoded data block, and a power consumption required for reading or writing the encoded data block in the memory, the encoded data block and the cost function being computed for different values of m,
- the memory (MEM) for storing the encoded data block corresponding to the value of m that minimizes the cost function.

7. A video decoder for decoding a bit-stream, said decoder comprising:

- a decoding unit (VLD,IQ,IT) for providing error data blocks from the bit-stream,
- a storage device (eENC-MEM) as claimed in claim 6 for storing encoded data blocks,
- means (eDEC-INT) for extracting at least one encoded data block from the memory and for decoding said at least one encoded data block so as to deliver a current motion-compensated data block (mc), and
- an adder for adding a current error data block (e) to the current motion-compensated data block, the output of said adder being provided to the input of the storage device.

8. A video encoder for encoding a sequence of pictures, a picture being divided into input data blocks, said encoder comprising:

- an encoding unit (T-Q) for providing a partially encoded data block,
- a decoding unit (IQ-IT) for providing a partially decoded data block from the partially encoded data block,
- a prediction unit (MEM-MC) comprising in series a storage device as claimed in claim 6 and means (eDEC-INT) for extracting at least one encoded data block from the memory and for decoding said at least one encoded data block so as to deliver a motion-compensated data block (mc),

- an adder for adding the motion-compensated data block to the partially decoded data block, the output of said adder being provided to the input of the prediction unit, and
- a subtracter for subtracting the motion-compensated data block from an input data block, the output of said subtracter being provided to the input of the encoding unit.

9. A portable apparatus comprising a storage device as claimed in claim 6.

10. A computer program product comprising program instructions for implementing, when said program is executed by a processor, a method as claimed in claim 1.

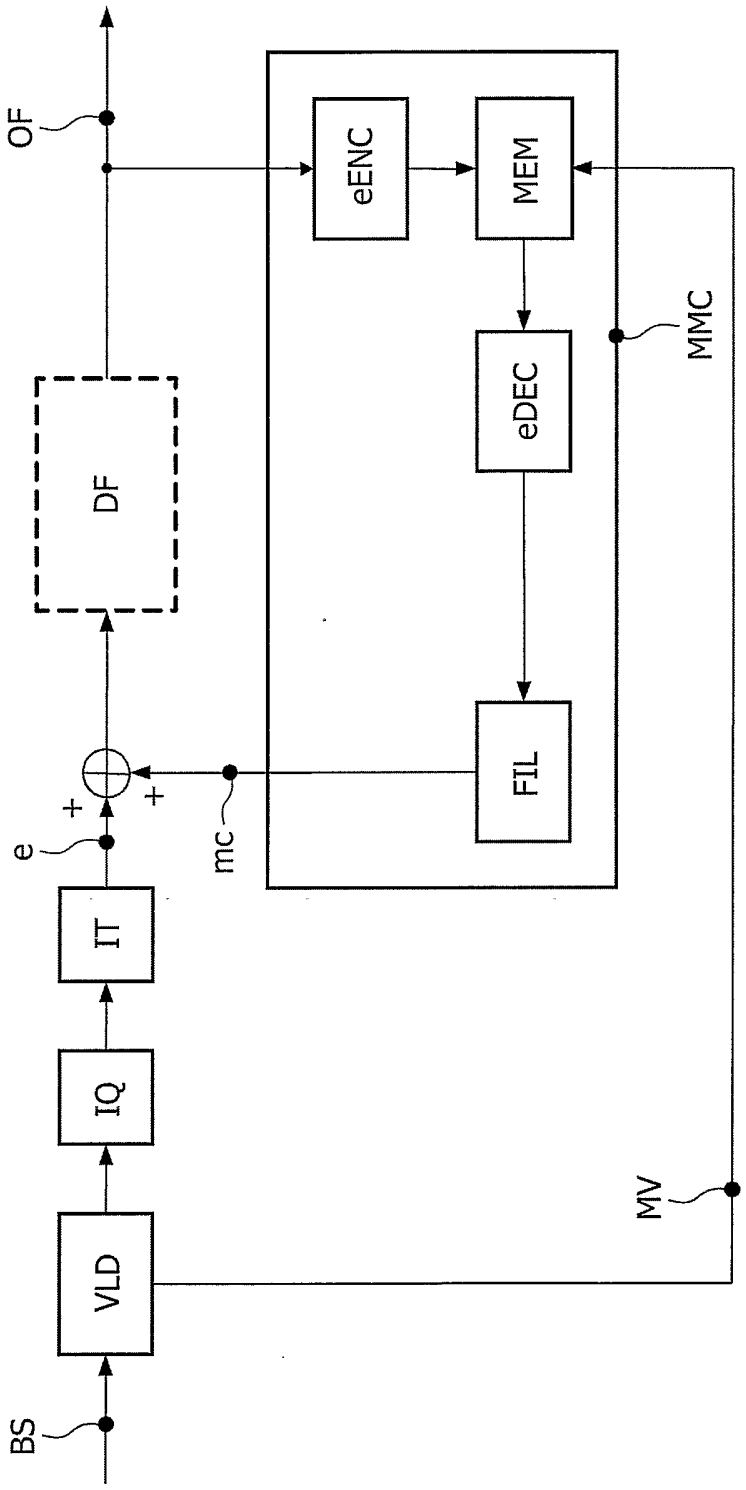


FIG.1

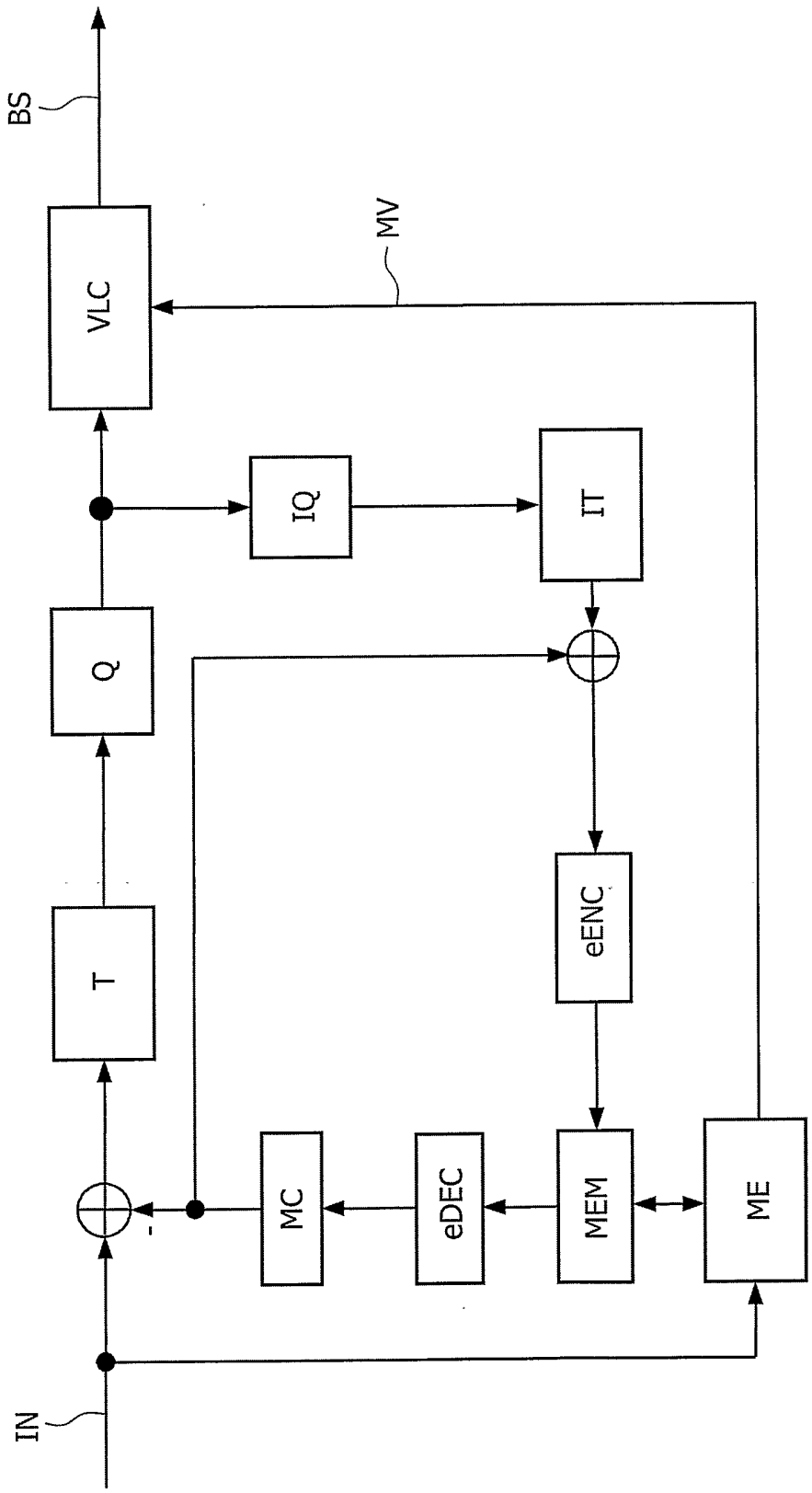


FIG.2

INTERNATIONAL SEARCH REPORT

IB2005/051845

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04N7/26 H04N7/36

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)
IPC 7 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, PAJ, INSPEC, COMPENDEX

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CHADDHA N ET AL: "A low power video encoder with power, memory and bandwidth scalability" VLSI DESIGN, 1996. PROCEEDINGS., NINTH INTERNATIONAL CONFERENCE ON BANGALORE, INDIA 3-6 JAN. 1996, LOS ALAMITOS, CA, USA,IEEE COMPUT. SOC, US, 3 January 1996 (1996-01-03), pages 358-363, XP010157168 ISBN: 0-8186-7228-5 the whole document ----- -/--	1-10

[X] Further documents are listed in the continuation of box C.

[ ] Patent family members are listed in 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	Date of mailing of the international search report
8 August 2005	30/08/2005

Name and mailing address of the ISA	Authorized officer
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	McGrath, S

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JUNG J ET AL: "Power Scalable Video Encoder for Mobile Devices based on Collocated Motion Estimation" PROCEEDINGS OF THE SPIE, SPIE, BELLINGHAM, VA, US, vol. 5308, 20 January 2004 (2004-01-20), pages 45-56, XP002310725 ISSN: 0277-786X the whole document	1-10
A	BOURGE A ET AL: "Low-power H.264 video decoder with graceful degradation" PROCEEDINGS OF THE SPIE, SPIE, BELLINGHAM, VA, US, vol. 5308, no. 1, 2004, pages 372-383, XP002334017 ISSN: 0277-786X cited in the application the whole document	1-10
A	XIAOAN LU ET AL: "Power efficient H.263 video transmission over wireless channels" PROCEEDINGS 2002 INTERNATIONAL CONFERENCE ON IMAGE PROCESSING. ICIP 2002. ROCHESTER, NY, SEPT. 22 - 25, 2002, INTERNATIONAL CONFERENCE ON IMAGE PROCESSING, NEW YORK, NY : IEEE, US, vol. VOL. 2 OF 3, 22 September 2002 (2002-09-22), pages 533-536, XP010607378 ISBN: 0-7803-7622-6 the whole document	1-10
A	VAN DER VLEUTEN R J ET AL: "Low-complexity scalable DCT image compression" IMAGE PROCESSING, 2000. PROCEEDINGS. 2000 INTERNATIONAL CONFERENCE ON SEPTEMBER 10-13, 2000, PISCATAWAY, NJ, USA,IEEE, vol. 3, 10 September 2000 (2000-09-10), pages 837-840, XP010529598 ISBN: 0-7803-6297-7 page 837, left-hand column, line 15 - right-hand column, line 26	3