(72) Inventors: VERMEIR, Thijs; Ieperstraat 151, B-8840 St-
aden (BE). SLOWACK, Jürgen; Borluutstraat 10, B-9140
Temse (BE).

[Continued on next page]

(54) Title: COMPLEXITY CONTROL OF VIDEO CODEC

(57) Abstract: There is provided means for controlling the computational
complexity related to video encoding. The invention comprises a dynamic
rate distortion cost error threshold and constrains the encoding complexity to
a given complexity target. The solution is selective and provides fast conver-
gence while incurring only a limited loss in rate distortion performance.

$$\sigma \leftarrow \sigma_0$$

$$equal\_sign \leftarrow True$$

$$function \text{ UPDATETHRESHOLD}(C_T, C_R)$$

$$\quad if (\sigma \geq 0) \neq (C_R \geq C_T) \ then$$

$$\quad\quad \sigma \leftarrow \sigma \ / -2$$

$$\quad\quad equal\_sign \leftarrow False$$

$$\quad else$$

$$\quad\quad if \ equal\_sign \ then$$

$$\quad\quad\quad \sigma \leftarrow 2\sigma$$

$$\quad\quad end \ if$$

$$\quad\quad equal\_sign \leftarrow True$$

$$\quad end \ if$$

$$\quad E_T \leftarrow \max(0, E_T + \sigma)$$

$$end \ function$$

Figure 3

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *of inventorship (Rule 4.17(iv))*

Published:

— *with international search report (Art. 21(3))*

1
COMPLEXITY CONTROL OF VIDEO CODEC

TECHNICAL FIELD

The present invention relates to methods, controllers and systems
5   for complexity control of video codec using coding units, as well
as software for implementing the methods or systems.

BACKGROUND

In the field of video coding and high efficient video coding, the
10   encoding application can be one of many processes running on the
host machine. Other processes can run in parallel, including
operating system processes, and even other encoders and/or
decoders. When the load on the host machine becomes significant,
the software processes may compete for the same resources and
15   influence each other's processing speed. This can be problematic,
particularly for a video encoder that is expected to deliver output
at regular time intervals. Although buffering may solve some of
these issues, adding a buffer introduces additional delay that is
in many cases undesired. Conventional solutions often comprise
20   carefully   fine   tuning   configuration   parameters   and/or
overprovisioning resources (e.g., selecting one of the more high-
end cloud instances) resulting in additional effort and increased
financial cost. Furthermore, when over dimensioning the system for
the statistical "worst case" (e.g., the maximum number of video
25   streams the cloud service should be able to handle), there could
be many occasions where the system resources are not being fully
used.

G. Correa et al. ("Complexity scalability for real-time HEVC
30   encoders," Journal of Real-Time Image Processing, pp. 1-16, 2014)
propose a complexity scalability method for HEVC in which CTUs are
dynamically constrained by limiting the prediction block (PB) and
setting a maximum tree depth for each CTU.

35   Zhao et al. ("Flexible mode selection and complexity allocation
in high efficiency video coding" Selected Topics in Signal
Processing, IEEE Journal of, vol. 7, no. 6, pp. 1135-1144, Dec

2013) propose a hierarchical complexity allocation scheme for HEVC based on linear programming.

While these methods provide alternative means for obtaining complexity control, they report high variation of the encoding time over multiple GOPs (Group Of Pictures).

SUMMARY OF THE INVENTION

It is an advantage of the present invention that a fully scalable solution for complexity control is provided, for optimal use of resources and where the spread in encoding time is converging.

Any of the embodiments of the invention can be implemented in an encoder.

An aspect of the present invention is a method or system or encoder to encode video frames having coding units which can be coded with a plurality of coding modes, by, for each coding unit of the frame the method steps of:

• Determining a rate distortion cost associated with encoding the coding unit with a given coding mode, for each of a chosen or selected number of coding modes being a reduced number of zones selected from the plurality of coding modes,

• Estimating an estimated rate distortion cost error as the difference between the lowest rate distortion cost amongst the determined rate distortion costs and an estimated lowest rate distortion cost amongst the complete plurality of coding modes,

wherein, the chosen number of coding modes is selected such that the difference between the rate distortion cost error and a target rate distortion cost error is minimised.

An example of a code is for example video compression.

Another aspect of the present invention relates to a

method or system or encoder for adaptive use of
image frame processing resources comprising defining
a rate distortion cost error threshold or target
associated with received image frames, determining a
5   difference between a complexity associated with the
rate distortion cost error threshold or target and a
threshold or target complexity, and updating the
rate distortion cost error threshold or target to
reduce this difference. Preferably, process of
10   reducing the difference should not involve a
significant increase in the use of resources.

In the above methods or systems or encoders the use of resources
may exceed or not exceed a target value. If more resources are used
15   than the target value then other processes can be impaired. If less
resources are used, then the quality of displayed video frames can
be reduced. Accordingly, continuous adaption to changing parameters
and video processing factors is preferred which results in the
image frame processing resources varying dynamically in time.
20

In one embodiment of the present invention there is provided
a system, a method or an encoder for adaptive use of image frame
processing resources for video compression, the image frame
processing resources varying dynamically in time, wherein the video
25   signal comprises image frames processed at a video rate, wherein
an image frame is expressed in a number of coding units, a coding
unit having a predefined maximum size, the method comprising the
steps: recursively splitting coding units into a quad-tree
structure of different coding units having levels and sub-levels,
30   wherein in a quad-tree structure, each level or sub-level is
recursively split into parts, a depth of a specific coding unit
being defined by the number of recursive splits of coding units
that have been used in order to reach the specific coding unit,
whereby the size of the specific coding unit is the maximum size
35   divided by the number of recursive splits, a coding unit at a
certain depth being encodable with a first number of different
coding modes. The rate distortion cost for a second number of

4

coding modes is determined, the second number being less or equal to the first number of different coding modes for the specific coding unit, the determining of the second number being such as to maximise the use of dynamically for the encoder available image frame processing resources, the difference between the rate distortion cost for the first and the rate distortion cost for the second number of coding modes is the rate distortion cost error, and the difference between the rate distortion cost error and a targeted rate distortion cost error has a minimum value for a certain coding mode, said coding mode is selected to encode the coding unit.

Alternatively a difference between an estimated rate distortion cost for the first number of coding modes and a determined rate distortion for the second number of coding modes is an estimated rate distortion cost error, and the difference between the estimated rate distortion cost error and a targeted rate distortion cost error has a minimum value for a certain coding mode, wherein said coding mode is selected to encode the coding unit.

The systems, encoders and methods described above provide means for maximizing the use of an available complexity budget (which refers, for example to a budget for vide frame processing resources) when performing video compression, while controlling the rate distortion cost.

A further useful step of embodiments of the present invention is to be able to normalize by dividing any of the processing factors with the number of pixels in the level to be calculated.

For example, the system, method or encoder can comprise use of a central processing or graphics processing unit to normalize the collection of pixels of the coding unit at level m and with index m, by dividing said coding unit with the number of pixels at the level m.

By normalizing the procedure, the system, encoder or method becomes independent of coding level.

An advantage of the present invention is to be able to keep inside
5   or close to a complexity budget per frame or to approach dynamically the complexity budget per frame. For example, the system, encoder or method can comprise, for each frame, the central processing or graphics processing unit providing a controller for adapting the threshold or target of the rate distortion cost error, based on
10   the difference between the reduced complexity and the value of the complexity target in a or the previous frame stored in a memory.

By having a complexity budget per frame, distortions can be smeared out over each frame.
15

An advantage of the present invention is to be able to calculate in elapsed CPU OR GPU time. For example, the system, encoder or method can comprise the central processing or graphics processing unit having the controller calculating the complexity for a frame
20   in elapsed central processing or graphics processing unit processing time for encoding the whole frame.

Accordingly, a practical measure of complexity can be to use the elapsed CPU OR GPU processing time. If too many resources are being
25   used then this can appear as an increased CPU OR GPU processing time. If too few resources are being used then this can appear as a decreased CPU OR GPU processing time.

An advantage of the present invention is to assume that the values
30   of α_m are independent of each other. For example, the system, encoder or method can comprising the central or graphics processing unit treating the actual number of coding modes for per coding unit being independent of each other.

35   In this way the system, encoder or method can be simplified by assuming that the coefficients α_m (which is the actual number of coding modes evaluated for a coding unit at level m) are independent

6

from each other. This simplification makes the calculations easier
but can result in errors. Accordingly, it is preferred if methods
of adaptive correction according to embodiments of the present
invention are used to dynamically adjust the video frame
processing.

An advantage of the present invention is to be able to estimate
$I_{m,N}$ (which is a rate distortion cost for all modes N). For example,
the system, encoder or method can comprise the central processing
or graphics processing unit estimating the rate distortion cost
comprising all coding modes by assuming said cost is proportional
to a linear model of the rate distortion cost comprising less than
all coding modes. By estimating using a model the calculation of
rate distortion costs for all coding modes is avoided.

Since the optimal rate distortion (RD) cost $I_{m,N}$ (for all modes N)
is not known for $I_{m,n}$ where $n < N$, (i.e. when the rate distortion
cost is not calculated for all coding modes N) it can be estimated,
for example to be proportional to $I_{m,n}$.

For example, the system, encoder or method can comprise that an
estimated rate distortion cost is linearly proportional to the rate
distortion cost comprising all coding modes multiplied by the
coefficient 0.87, 0.92, 0.94, 0.95 for the depth 0, 1, 2 and 3
respectively.

The inventors have found that the above mentioned coefficients
provide a good estimate.

An advantage of the present invention is to be able to estimate
only two groups of modes instead of many modes. For example, the
system, encoder or method can comprise the central processing or
graphics processing unit evaluating two groups of modes, the first
group comprising the 2Nx2N merge mode and the second mode
comprising all other modes. Accordingly, the system, encoder or
method can be simplified by assuming that there are two groups of
coding modes.

7

An advantage of the present invention is to be able to program a
controller. For example, the system, encoder or method can
comprise the central processing unit or graphics processing unit
5   implementing the controller based on a root-finding method. There
are two types of root finding, namely bracketed and open methods.
Both bracketed and open methods can be used with embodiments of
the present invention. Different methods will have different
performance characteristic, for example bracketed methods will
10   converge but may take time to do so. Open methods may be quicker
but do not necessarily always have a guarantee of convergence.

Examples of root finding methods such as a bisection method, a
false position method, the Brent Dekker method, Muller's method,
15   Simple fixed point iteration method, Newton's method,
multidimensional Newton's method, secant method or similar.

A controller can be implemented to try out various modes for each
coding unit; whereby the controller can be based on a root finding
20   method such as the root-finding bisection method or one of the
other methods mentioned above.

For example, the system, encoder or method can comprise the central
or graphics processing unit instructing the controller to execute
25   a root finding method of those mentioned above such as a bisection
method with the following steps:

Set initial value to the step size, assign a variable "equal_sign"
to True
30   If A and B can each be true or false, and A is the condition that
the step size is larger than or equal to zero, and B is the
condition that the reduced complexity is larger than or equal to
the complexity threshold, and A is different from B, then
Reduce the step size to half and change direction
35   Set variable "equal_sign" to false
If A is equal to B and variable "equal_sign" is true, then double
the step size without changing direction,

Set variable "equal_sign" to false, and let the rate distortion cost error be increased by one step size unless this is smaller than zero, then let the rate distortion cost error target be zero.

This may be written as pseudocode (Figure 3):

$\sigma \leftarrow \sigma_0$

$equal\_sign \leftarrow True$

$function \; \text{UPDATETHRESHOLD}(C_T, C_R)$

$\quad if(\sigma \geq 0) \; \neq (C_R \geq C_T) \, then$

$\quad\quad \sigma \leftarrow \sigma \; / -2$

$\quad\quad equal\_sign \leftarrow False$

$\quad else$

$\quad\quad if \; equal\_sign \; then$

$\quad\quad\quad \sigma \leftarrow 2\sigma$

$\quad\quad end \, if$

$\quad\quad equal\_sign \leftarrow True$

$\quad end \, if$

$\quad E_T \leftarrow \max(0, E_T + \sigma)$

$end \; function$

This can be understood as

1. Assign the step size sigma to an initial value, sigma zero.
2. Assign the variable "equal_sign" to True. (Can also be assigned to False, it may just change the number of iterations).
3. The function UPDATE…. is just a notation/name for the rest of the code.
4. A first condition A is that the step size is larger or equal to zero. A condition B is that the reduced complexity CR should be larger or equal to the complexity threshold CT. If A is not equal to B, e.g (A,B) is (True,False) or (False,True) then
5. Assign the step size sigma to sigma divided by -2, thus reducing the step size to half and changing direction.
6. Assign "equal_sign" to False

7. Else: if the condition in 4 is not fulfilled

8. And if "equal_sign" is True

9. Then double the step size sigma (in the same direction)

10.   Endif for line 8

11.   Assign "equal_sign" to False (in any case)

12.   Endif of line 4

13.   Assign RD cost error target ET to (the RD cost error
        target ET + step size sigma) or zero, in case the (the RD
        cost error target ET + step size sigma) is less than zero

14.   End of whole function.

The controller can be implemented with the above root-finding
bisection method or another root finding method like the ones
described above. The bisection method involves a low complexity
and allows the methods, encoders and systems according to
embodiments of the present invention to adapt dynamically and
continuously.

In another embodiment there is provided a system, an encoder or a
controller, or a method for adaptive use of image frame processing
resources comprising a central processing or graphics processing
unit connected to at least one memory, the central processing or
graphics processing unit being configured to receive frames of a
video signal, the central processing or graphics processing unit
defining a rate distortion cost error threshold or target, the
difference between the corresponding first complexity achieved with
the use of the rate distortion cost error threshold or target and
a first complexity threshold or target is stored in the memory as
a reference for the calculation of an updated rate distortion cost
error threshold or target, wherein the difference between a second
complexity achieved with the updated rate distortion cost error
threshold or target, and the former first complexity threshold or
target is smaller or equal to the difference between the former
first complexity and the first complexity threshold or target. This
results in the same difference compared to the complexity threshold
or target or a better performance.

In another embodiment there is provided a system, an encoder or a controller, or a method for adaptive use of image frame processing resources comprising a central processing or graphics processing unit connected to at least one memory, the central processing or graphics processing unit being configured to receive frames of a video signal, each frame being split into coding units comprising pixels, the central processing or graphics processing unit defining a rate distortion cost error threshold, the difference between the corresponding complexity and a complexity threshold is stored in the memory as a reference for the calculation of an updated rate distortion cost error threshold, wherein the difference between the complexity resulting from the updated rate distortion cost error threshold, and the former complexity threshold is smaller or equal to the difference between the former complexity and the complexity threshold.

The system, encoder or controller provides means or a method step for maximizing the use of an available complexity budget when performing video compression, while controlling the rate distortion cost.

For example, the system, encoder, controller or method can comprise a coding unit at level m with an index m being normalised by instructing the central processing or graphics processing unit to divide it with the number of pixels at the level m. The controller can be adapted to perform the above mentioned normalization.

The normalization has the advantage of making the system, encoder, controller or method independent of coding level.

An advantage of the present invention is that the system, encoder or controller or method keeps within the complexity budget per frame. For example, the system, encoder controller or method can comprise for each frame, the threshold or target of the rate distortion cost error being adapted by the central processing or graphics processing unit, based on the difference between a desired reduced complexity and the complexity target in the previous frame.

The controller can be adapted to perform the above mentioned adaptation.

By having a complexity budget per frame, distortions can be smeared out over each frame.

An advantage of the present invention is to be able to calculate in elapsed CPU OR GPU time. For example, the system, encoder, controller or method can provide that the complexity for a frame is expressed in elapsed central processing or graphics processing unit processing time for encoding the whole frame. The controller can be adapted to perform the above mentioned expression.

A practical measure of complexity can be to use the elapsed CPU OR GPU time.

An advantage of the present invention is to be able to assume that the values of α_m (see above) are independent of each other. For example, the system, encoder, or method can comprise the central processing or graphics processing unit being instructed to consider the actual number of coding modes per coding unit as being independent of each other. The controller can be adapted to implement the above mentioned independency.

The system, encoder controller or method can be simplified by assuming that the coefficients α_m are independent.

An advantage of the present invention is to be able to estimating the $I\_(m,N)$ (see above). For example, the system, encoder, controller or method can provide that the rate distortion cost (comprising all coding modes) is estimated by the central processing or graphics processing unit to be proportional to a linear model of the rate distortion cost comprising less than all coding modes. The controller can be adapted to perform the above mentioned estimation.

Since the optimal RD cost I_(m,N) (for all modes N) is not known for I_(m,n) where n<N, it can be estimated, for example to be proportional to I_(m,n).

An advantage of the present invention is to be able to rely on linear dependency. For example, the system, encoder or method can provide that the central processing or graphics processing unit is instructed to consider an estimated rate distortion cost being linearly proportional to the rate distortion cost comprising all coding modes, multiplied by the coefficient 0.87, 0.92, 0.94, 0.95 for the depth 0, 1, 2 and 3 respectively.

The inventors have found that the above mentioned coefficients provide a good estimate.

An advantage of the present invention is to be able to estimate only two groups of modes instead of many modes. For example, the system, encoder, controller or method can comprise that only two groups of modes are evaluated by the central processing or graphics processing unit, the first group comprising the 2Nx2N merge mode and the second mode comprising all other modes. The controller can be adapted to perform the above mentioned evaluation.

The system, encoder, controller or method can be simplified by assuming that there are two groups of coding modes.

In another aspect the present invention provides a controller for adaptive use of image frame processing resources, the controller comprising a central processing unit connected to at least one memory, the central processing unit being configured to receive frames of a video signal, each frame being split into coding units comprising pixels, the central processing unit defining a first rate distortion cost error threshold or target, the difference between the corresponding complexity achieved with the first rate distortion cost error threshold or target and a first complexity threshold is stored in the memory as a reference for the calculation of an updated rate distortion cost error threshold or target,

wherein the difference between a second complexity resulting from use of the updated rate distortion cost error threshold, and the former first complexity threshold or target is smaller or equal to the difference between the former first complexity and the first complexity threshold or target.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 shows an embodiment of the present invention comprising a scatter plot of measurements of $I_{m,n}$ and $I_{m,N}$.

Figure 2 shows an embodiment of the present invention comprising a Reduced encoding time C_R per frame.

Figure 3 show an embodiment of the present invention comprising a pseudo code of the proposed complexity controller

Figure 4 shows an embodiment of the present invention comprising a Encoding time per frame for (a) the anchor encoder and (b)

the complexity constrained encoder and (c)

the average encoding time (ET), PSNR and bitrate.

Figure 5 shows an embodiment of the present invention comprising RD curves for different complexity thresholds (CT): (a) Kimono (b) ParkScene, (c) Comparison of encoding time (ET) and BD-rate (BDR) for different complexity thresholds.

Figure 6 shows an embodiment of the present invention for implementing a system.

Figure 7 shows an embodiment of the present invention comprising a flow diagram.

DEFINITIONS

An "image frame" or "video frame" or just "frame" can be an individual image. A sequence of such frames e.g. images shown during certain duration of time, can provide the appearance of motion. A frame can comprise a collection of pixels.

A frame can be divided into "blocks" or "Coding Tree Units" (CTUs). CTUs can be recursively sub-divided into smaller "Coding Units" (CUs).

A "Prediction Unit" (PU) can be a collection of pixels for which a prediction can be generated.

14

A "Transform Unit" (TU) can be a collection of pixels for which a transformation can be applied, e.g. cosine, sine, Fourier.

The "Lagrange multiplier" or "Lagrangian" is factor that can be used in a mathematical method to solve optimization problems.

5 A "Motion Vector" (MV) is a vector that can be used to signal a block or unit in an image frame based on the position of this block or unit (or a similar one) in another image frame, called the reference (image) frame.

A "SKIP" or "merge SKIP" can be used in HEVC resulting in that no

10 residue is transmitted as codec.

"Complexity" can be used as a measure for the amount of occupied resources. A resource refers to a digital processing resource especially a video frame digital processing resource such as provided by a processing engine and memory. Examples of a

15 processing engine are a microprocessor, an FPGA, an ASIC, a graphics card, a GPU or a CPU. For example, the time take for a processing engine such as a CPU or a GPU to execute a process can be used as an indication of the amount of resources used.

A "coding mode" can be a method used to provide the coded version

20 of a group of pixels, e.g. a block or a unit.

"Quantization Parameter", QP, controls the trade-off between bit rate and quality.

A "target" or "target value or range" or a "threshold value" can be defined more generally as a "performance indicator". A target

25 value is a value that ideally is to be reached. Actually achieved results may go above or below a target value. A threshold value sets a limit for a desired performance. Actually achieved results are poor when the threshold is not achieved and are generally good when the threshold value has been achieved. A target value may be

30 defined by upper and lower threshold values i.e. the meaning of target includes a target range, whereby being inside the range, i.e. between the threshold values means that a satisfactory result has been achieved. Providing target values with upper and lower thresholds can reduce the jitter created by dynamic adaption

35 algorithms being activated when the difference between the target value and the achieved value is small. Hence, thresholding and targets or target ranges are related performance indicators and in

the description and claims threshold and target are similar, e.g. interchangeable depending on design choice.

DETAILED DESCRIPTION

5   The present invention will be described with respect to particular embodiments and with reference to certain drawings but the invention is not limited thereto but only by the claims. The drawings described are only schematic and are non-limiting.

10   Furthermore, the terms first, second, third and the like in the description and in the claims, are used for distinguishing between similar elements and not necessarily for describing a sequential or chronological order. The terms are interchangeable under appropriate circumstances and the embodiments of the invention can

15   operate in other sequences than described or illustrated herein.

The term "comprising", used in the claims, should not be interpreted as being restricted to the means listed thereafter; it

20   does not exclude other elements or steps. It needs to be interpreted as specifying the presence of the stated features, integers, steps or components as referred to, but does not preclude the presence or addition of one or more other features, integers, steps or components, or groups thereof. Thus, the scope of the expression

25   "a device comprising means A and B" should not be limited to devices consisting only of components A and B. It means that with respect to the present invention, the only relevant components of the device are A and B. Similarly, it is to be noticed that the term "coupled", also used in the description or claims, should not be

30   interpreted as being restricted to direct connections only. Thus, the scope of the expression "a device A coupled to a device B" should not be limited to devices or systems wherein an output of device A is directly connected to an input of device B. It means that there exists a path between an output of A and an input of B

35   which may be a path including other devices or means.

Elements or parts of the described devices may comprise logic

encoded in media for performing any kind of information processing. Logic may comprise software encoded in a disk or other computer-readable medium and/or instructions encoded in an application specific integrated circuit (ASIC), field programmable gate array

5   (FPGA),   or   other   processor   or   hardware.

In the video compression standard "High Efficiency Video Codec" (HEVC, also referred to as h.265) an image frame can be expressed in a number of "coding tree units", CTUs, where the maximum size

10   of a CTU can be set to a number of pixels, the maximum size of a CTU can be set to 64x64 pixels. A CTU can be recursively split into a quad-tree structure of different "coding units" (CUs). In a quad-tree structure, each sub-level can recursively be split into a number of parts such as four parts. The depth of a CU is defined

15   by the number of recursive splits of its CTU that have to be performed in order to reach it. For example, a CU with depth zero (0) is not split and has a maximum size such as 64x64 pixels and a CU with depth three (3) is split three times and has the size such as 8x8 pixels (which is, for example, the minimum size

20   according to the HEVC standard). To find the optimal encoding mode for a CTU, a cost of different modes such as the Lagrangian cost of different modes can be recursively evaluated and compared.

In video compression, one way of reducing the amount of data to be

25   handled is to use prediction. Motion vectors are used in inter-frame prediction, i.e. between frames of different points in time. A "Motion Vector" (MV) is a vector that can be used to signal (e.g. provide metadata for) a block or unit in an image frame based on the position of this block or unit (or a similar one) in another

30   image frame, called the reference (image) frame. The methods "2Nx2N merge mode" and "Advanced Motion Vector Prediction" (AMVP) can be defined to signal motion vectors (MV), e.g., to provide metadata on the blocks or units to the decoder

35   *Defining a complexity budget, target or threshold*
     If a digital process is working on data and additional resources suddenly become available, it could be beneficial if the digital

process could access these resources to e.g. to obtain improved quality. For example, in cases where the load on the host computer could change dynamically, and/or the same software is expected to run flexibly on multiple hardware platforms (e.g., cloud instances of different providers), it could be of advantage if the digital processes have an adaptive behaviour.

In one embodiment of the present invention there can be a complexity budget or target assigned to each digital process. There can be means provided for having a or each process dynamically reconfiguring itself to respect this budget, for example by using as much of the budget as possible, without exceeding it but preferably at the same time not using considerably less of the budget which could result in poorer quality displayed video.

Consider a frame being divided into M CUs so that for each CTU there can be maximum of $1 + 2^2 + 2^4 + 2^8$ CUs. A particular CU at index or level m, $m \in \{1, \ldots, M\}$ can be encoded using n different coding modes (or methods), $n \in \{1, \ldots, N\}$. N may be different at different CU depths and/or CU locations. CUs at a larger depth (the larger depth meaning further in the sub-branches) will only be evaluated if the parent of the CU evaluates a need to split further. If this is the case, the CU is added to the list of evaluated CUs, said list having a length Ω. For the n'th mode with distortion $D_{m,n}$ (related to loss of image quality) and rate $R_{m,n}$ (related to the amount of data needed), a rate distortion costs such as a Lagrangian "rate distortion cost" $J_{m,n}$ can be defined as:

$$J_{m,n} = D_{m,n} + \lambda R_{m,n} \tag{1}$$

where λ is the Lagrange multiplier. In general, the rate distortion cost can be defined with other means than the Lagrange multiplier. In a typical encoder, only the coding mode with the lowest rate distortion cost will actually be used for encoding. If $I_{m,n}$ is the optimal RD cost for a CU at index m after evaluating n modes:

$$I_{m,n} = \min(J_{m,1}, \dots, J_{m,n}) \tag{2}$$

the computational complexity required for evaluating mode n for a given CU with index m will be denoted $C_{m,n}$. When evaluating all CU configurations and coding modes, the total frame complexity is given by $\sum_{m=1}^{\Omega} \sum_{n=1}^{N} C_{m,n}$.

Defining and simplifying the optimization problem

In one embodiment, the complexity is reduced by not evaluating all encoding modes n per level m. Let α_m be the actual number of encoding modes evaluated for a CU at index m $(\alpha_m \in \{1, \dots, N\})$ Note that the value of α_m can have an influence on the list of evaluated CUs and therefore can change Ω.

Clearly, when only a limited set of encoding modes is evaluated (i.e., with $\alpha_m < N$) there is a possible penalty in terms of RD cost. This penalty will be called the RD cost error E_{m,α_m}:

$$E_{m,\alpha_m} = I_{m,\alpha_m} - I_{m,N} \tag{3}$$

The RD cost error can be zero if the optimal encoding mode is found within the first encoding modes, otherwise E_{m,α_m} will be positive. The α_ms can be defined so that rate and distortion are optimized while the reduced complexity f (per frame) does not exceed a particular complexity threshold or target C_T. In other words:

$$C_R = \sum_{m=1}^{\Omega} \sum_{n=1}^{\alpha_m} C_{m,n} \tag{4}$$

$$\min_{\alpha_0, \dots, \alpha_\Omega} \sum_{m=1}^{\Omega} E_{m,\alpha_m} \ \textit{subject to:} \ C_R \le C_T \tag{5}$$

Solving this constrained optimization problem is complex, particularly since the RD cost (and therefore also I_{m,α_m} and E_{m,α_m}) in general depends on decisions taken in the context of previously coded CUs. If the number of encoding modes to evaluate has been severely restricted for a particular CU, the RD performance for that CU could be relatively low, which increases the RD cost for

19

spatially and temporally neighbouring CUs trying to exploit correlation with this CU.

In one embodiment of the present invention the optimization problem
5   can be simplified with certain assumptions. The first assumption is that the value of α_m for a CU can be decided independently from other CUs. In other words, it can be assumed that the α_ms are independent from each other as well as the resulting E_{m,α_m}s. A second assumption is that distributing the RD cost error E_{m,α_m}
10 equally over the CUs results in improved human visual perception. So to cope with different CU depths, the RD cost error can be normalized per level, for example by dividing it with the number of pixels at the specific level. $\bar{E}_T$ can then denote the cost error threshold or target per pixel and $\bar{E}_{m,\alpha_m}$ can denote the cost error
15 per pixel. Then Eq. (5) can be reformulated as:

$$\min_{\alpha_0,\dots,\alpha_\Omega} \sum_{m=1}^{\Omega} |\bar{E}_T - \bar{E}_{m,\alpha_m}| \; subject\; to\colon C_R \leq C_T \tag{6}$$

Note that the expression in (6) is fully parameterized and can be
20 used for any CU in the frame.

Based on equation (6) a complexity system, encoder, controller or method can be implemented where the targeted error cost or threshold error cost $\bar{E}_T$ can be varied frame-by-frame, striving to
25 match the complexity target C_T, i.e. to bring C_R as close as possible (or equal to) C_T but without exceeding it.

Exemplary embodiments
Now follows some exemplary embodiments of the present invention.
30 In one embodiment, the number of encoding modes can be assumed to be simplified to two groups (instead of the complete number of N modes) for illustration purposes. However, this should not be construed as a limitation of the invention as such. The two groups used can be respectively the 2Nx2N merge mode and another group
35 containing all other encoding modes.

Deciding when to stop encoding

A system, encoder, controller, encoder or method can estimate $\bar{E}_{m,\alpha_m}$ to decide whether or not to continue evaluating other modes, where

5    $\bar{E}_{m,\alpha_m}$ is a normalized version of Eq. (3). While $I_{m,n}$ is available at mode n, $I_{m,N}$ is not available and therefore needs to be estimated. Figure 1 illustrates the relationship between $I_{m,n}$ and $I_{m,N}$ for depth or level 0 after a 2Nx2N merge mode. Due to efficient construction of the merge candidate list, there is a high probability that 2Nx2N

10   merge mode is the optimal encoding mode or has low RD cost difference with the optimal mode. So, high correlation is expected. Therefore, in one embodiment, a simple linear model can be used to predict $I_{m,N}$. The coefficients of the linear model for depth {0, 1, 2, 3} can be respectively {0.87, 0.92, 0.94, 0.95}. A higher

15   coefficient indicates that on average a lower $\bar{E}_{m,\alpha_m}$ is expected if further evaluation of encoding modes is not done.

Designing the complexity controller

Embodiments of the present invention include a complexity

20   controller which can be implemented as a standalone device, or embedded for example in a system or encoder. The goal of the complexity controller can comprise the respect of the complexity target C_T. Figure 2 shows a relation between C_R and $\bar{E}_T$. An anchor (or reference) encoder can be used to extract I_{m,α_m} and the encoding

25   is re-run with multiple targets or thresholds $\bar{E}_T$. When $\bar{E}_T = 0$ (i.e. no RD cost error in the frame), the complexity is reduced only on the CUs where the 2Nx2N merge mode was the optimal mode. When $\bar{E}_T \geq 0$ different CUs will stop encoding after mode α_m when I_{m,α_m} is below the normalized threshold or target. The complexity reaches a

30   minimum when all CUs (at depth 0) only evaluate the 2Nx2N merge mode, whereby further increasing of the threshold or target will not decrease the complexity anymore. With a prediction model, $\bar{E}_T$ can be empirically defined for a constrained complexity C_T. But, this model or relation is not available for the complexity

35   controller in a real-time implementation.

However, C_R and $\bar{E}_T$ are respectively an increasing and a decreasing monotonic function of α_m (see Eq.5). So, increasing the number of evaluated encoding modes α_m will always increase C_R and decrease $\bar{E}_T$. Therefore, the complexity controller can use $\bar{E}_T$ to control the frame complexity C_R, due do the properties of monotonic functions. This is a typical root finding problem. However, there are additional difficulties compared to conventional root finding algorithms. In the present embodiment, while $\bar{E}_T$ is a constant for a frame, it does change slightly every frame due to various video processing factors such as the changing complexity of the video or inaccurate measurements of the encoding complexity. Therefore, a complexity controller that adapts $\bar{E}_T$ continuously on a frame-by-frame basis is required.

In a conventional root-finding problem the algorithm finds a point closer to the root at each iteration. In the present embodiment, only a single iteration can be used each frame and the algorithm can fine-tune this value on the next frame. Therefore, the basic complexity controller here presented can be based on a root finding method such as the root-finding bisection method. This proposed method is explained with pseudo-code in Figure 3. If the root is outside the current interval, σ will be doubled if the previous iteration was in the same direction (i.e. σ has the same sign). Doubling allows the method to search faster for the correct interval. When the interval with the root is found, it is divided by -2 so that the direction is changed. This can allow more precise approximation of the correct value while still being able adapt to local changes in the video or noise on the complexity measurement. Alternative root finding methods can be used. There are two types of root finding, namely bracketed and open methods. Both bracketed and open methods can be used with embodiments of the present invention. Different root finding methods will have different performance characteristics, for example bracketed methods will converge but may take time to do so. Open methods may be quicker but do not necessarily always

have a guarantee of convergence. Examples of root finding methods
included the bisection method, a false position method, the Brent
Dekker method, Muller's method, Simple fixed point iteration
method, Newton's method, multidimensional Newton's method, secant
5   method etc.

Finally, $\bar{E}_T$ is incremented with the current σ value. As $\bar{E}_T = 0$
indicates maximum complexity, it is not useful to accept negative
values for $\bar{E}_T$. Note that σ and *equal_sign* can be initialized with
10   any value (e.g. respectively σ_0 and *True*), this will only have
influence on the iterations needed to find the correct E_F value.

Experimental results
The proposed system, encoder or method can be evaluated using
15   sequences from class B described in the common test conditions of
HEVC (F. Bossen, "Doc. JCTVC-L1100: Common test conditions and
software reference configurations," JCTVC, Geneva, Switzerland,
Tech. Rep., January 2013), i.e. Kimono, ParkScene, Cactus,
BQTerrace, BasketballDrive. The HM 16.2 reference encoder is used
20   as the anchor or reference. There is focus on low delay settings
(i.e. main tier and P slices only), with QP values 22, 27, 32, 37
and open GOP (Group Of Pictures) structure. Figure 4 shows the
encoding time per frame for QP=22, for the anchor (Figure 4(a))
and the proposed method (Figure 4(b)). For the anchor, although
25   encoding settings are identical, there is a significant
difference in encoding time (i.e., 42.8%) between Cactus (highest
encoding time) and Kimono (lowest encoding time). The frame with
the highest complexity in the BasketballDrive sequence has 14.1%
more encoding time compared to the frame with the lowest
30   complexity. For the proposed method (Figure 4(b)), a complexity
target of 40 seconds per frame was defined. As can be observed in
the figure, the encoder initializes during the first frames in an
attempt to find the RD cost threshold or target using the
bisection method or another root finding method. There are two
35   types of root finding, namely bracketed and open methods. Both
bracketed and open methods can be used with embodiments of the

present invention. Different root finding methods will have
different performance characteristics, for example bracketed
methods will converge but may take time to do so. Open methods
may be quicker but do not necessarily always have a guarantee of

5   convergence. Examples of root finding methods include the
bisection method, a false position method, the Brent Dekker
method, Muller's method, Simple fixed point iteration method,
Newton's method, multidimensional Newton's method, secant method
etc.

10

After this initialization phase, the complexity controller adapts
the threshold or target frame-by-frame, and manages to keep
computational complexity close to the complexity budget, with an
average deviation of 0.57 seconds (1.4%). Reduced complexity

15   comes at a limited cost, e.g., 0.13 dB PSNR loss and 1.6% bitrate
loss for a complexity reduction of 48% for the Cactus sequence.

Figure 5 shows RD curves for different complexity targets (CT),
for Kimono (Figure 5(a)) and ParkScene (Figure 5(b)). The CT is

20   defined relative to the anchor encoding time at same QP point
(anchor encoding time for each QP are provided in the Figure). For
a complexity target of 80%, the BD-rate (BD = BjØntegaard Delta
Rate) increase ranges from 0:00% (for ParkScene) to 1:72% (for
BqTerrace).

25

The proposed system, encoder or method for complexity control is
compared with a number of fixed complexity methods incorporated in
the HM encoder. Using early skip detection (ESD), an average
complexity of 74:8% (relative to the anchor) is measured. When only

30   a maximum CU depth of 2 (MD2) is allowed, an average encoding time
of 41:8% of the anchor encoding is measured. At high complexity
targets, low BD-rate loss is measured compared to the ESD method.
The BD-rate loss at CT = 40% is lower for the BasketBallDrive,
Cactus and ParkScene sequences. Embodiments of the present

35   invention provide complexity control, and in general have better
RD performance compared to the ESD and MD2 methods. Occasionally,
the latter methods show better performance, indicating that there

24

is still room for improvement. Improvements can include, for example, improving the RD cost error prediction and/or allowing more groups of encoding modes instead of only 2Nx2N merge mode.

5   Figure 6 shows an illustration of an embodiment of the present invention. The central processing unit or graphics processing unit 10 can be connected to and communicate with a memory 11, a non-volatile memory such as a solid state memory or a hard drive 12 and a network 13, e.g. to form a system or encoder or to carry out

10   a method for adaptive use of image frame processing resources. The system or encoder can comprise a central processing or graphics processing unit connected to at least one memory, the central processing or graphics processing unit being configured to receive frames of a video signal, each frame being split into coding units

15   comprising pixels, the central processing or graphics unit defining a first rate distortion cost error threshold or target, the difference between the corresponding first complexity achieved with this rate distortion cost error threshold or target and a first complexity threshold or target is stored in the memory as a

20   reference for the calculation of an updated rate distortion cost error threshold, wherein the difference between a second complexity resulting from use of the updated rate distortion cost error threshold, and the former first complexity threshold or target is smaller or equal to the difference between the former first

25   complexity and the first complexity threshold or target.

A coding unit at level m and with index m can be normalised by instructing the central processing or graphics unit to divide it with the number of pixels at the level m.

30

For each frame, the threshold or target value or the rate distortion cost error can be adapted by the central processing or graphics processing unit, based on the difference between the reduced complexity and the complexity target in the previous frame.

35

The complexity for a frame can be expressed in elapsed central processing or graphics processing unit processing time for encoding the whole frame.

5   The central processing unit or graphics processing can be instructed to consider the actual number of coding modes for per coding unit as independent of each other.

The rate distortion cost comprising all coding modes can be
10   estimated by the central processing or graphics processing unit to be proportional to a linear model of the rate distortion cost comprising less than all coding modes.

The central processing or graphics processing unit can be
15   instructed to consider an estimated rate distortion cost as being linearly proportional to the rate distortion cost comprising all coding modes, multiplied by the coefficient 0.87, 0.92, 0.94, 0.95 for the depth 0, 1, 2 and 3 respectively.

20   Only two groups of modes need be evaluated by the central processing or graphics processing unit, the first group comprising the 2Nx2N merge mode and the second mode comprising all other modes.

The central processing or graphics processing unit can implement
25   a controller by means of a root-finding method such as the bisection method. Other root finding methods can be used. There are two types of root finding, namely bracketed and open methods. Both bracketed and open methods can be used with embodiments of the present invention. Different root finding methods will have
30   different performance characteristics, for example bracketed methods will converge but may take time to do so. Open methods may be quicker but do not necessarily always have a guarantee of convergence. Examples of root finding methods such as a bisection method include a false position method, the Brent Dekker method,
35   Muller's method, Simple fixed point iteration method, Newton's method, multidimensional Newton's method, secant method etc.

Figure 7 shows a flow chart for an embodiment of the present invention. In step 20 the various parameters are being initialized (for example, the complexity CR can be set to 0) and the value can be stored in the memory 11 (alternatively for the memory 11 the

5   hard drive 12 can be used for storage throughout this section). In step 21 the CPU or GPU 10 can select a coding unit n from a list of all possible coding units stored in the memory 11. In step 21 the CPU or GPU 10 can select a coding mode n from a list of coding modes that can be stored in the memory 11. In step 23 the rate

10   distortion cost I can be calculated. In step 24 the CPU or GPU calculates the normalized rate distortion cost error and compares the result with a targeted normalized rate distortion cost error that can be stored in memory 11. The normalization can be made by dividing the RD cost error of CU n with the number of pixels in

15   the CU n. If the difference between the calculated RD cost error and the referenced RD cost error (for CU m and mode n) is higher than the targeted difference, the CPU or GPU returns to step 22 to select the next coding mode and perform calculations in steps 23 and 24. If the calculated difference is lower or equal to the

20   targeted difference, the coding mode n is selected. The CPU or GPU now returns to step 21 and selects the next CU m. When a coding mode has been chosen for to all CUs, the frame has been encoded and can be transmitted. In practice, the RD cost can be provided by the encoder. A common way to measure the complexity can be to

25   measure the total CPU or GPU time needed to encode the full frame. If this is used, the used complexity can be compared with the budgeted complexity when the whole frame has been encoded. A controller can then be implemented to adjust the targeted RD cost error between frames in order to keep within the complexity budget

30   or to approach it with time. Encoded images can be transmitted to a network for example.

In accordance with an embodiment of the present invention, an encoding engine is provided for encoding images from at least one

35   input source. The encoding engine can be included in the controller for example. The encoding engine can include one or more FPGA, ASIC, or microprocessors, processors, controllers, or the central

processing unit (CPU) 10 and/or a Graphics Processing Unit (GPU), and can be adapted to carry out functions by being programmed with software, i.e. one or more computer programmes.

5   The encoding engine may have the memory 11 (such as non-transitory computer readable medium, RAM and/or ROM), an operating system running on a microprocessor, optionally a display such as a fixed format display, data entry devices such as a keyboard, a pointer device such as a "mouse", serial or parallel ports such as I/O

10   ports to communicate with other devices, or network cards and connections to connect to any of the networks or to peripheral devices.

In accordance with another embodiment of the present invention

15   software may be implemented as a computer program product which has been compiled for a processing engine in the encoding engine described above, e.g. for maximizing the use of a complexity budget, wherein expression is parameterized and valid for all coding level. The computer program product may be stored on a non-

20   transitory signal storage medium such as an optical disk (CD-ROM or DVD-ROM), a digital magnetic tape, a magnetic disk, a solid state memory such as a USB flash memory, a ROM, etc.

The encoding software can be embodied in a computer program product

25   adapted to carry out the following functions when the software is loaded onto the respective device or devices and executed on one or more processing engines such as microprocessors, FPGA, ASIC etc:

adaptive use of image frame processing resources for video

30   compression, the image frame processing resources varying dynamically in time, wherein the video signal comprises image frames processed at a video rate, wherein an image frame is expressed in a number of coding units, a coding unit having a predefined maximum size and a controller receiving said image

35   frame, and/or recursively splitting coding units into a quad-tree structure of different coding units having levels and sub-levels, wherein in a quad-tree structure, each level or sub-level is

recursively split into parts, a depth of a specific coding unit being defined by the number of recursive splits of coding units that have been used in order to reach the specific coding unit, whereby the size of the specific coding unit is the maximum size divided by the number of recursive splits.

The encoding software can be embodied in a computer program product adapted to carry out the following functions when the software is loaded onto the respective device or devices and executed on one or more processing engines such as microprocessors, etc:

wherein a coding unit at a certain depth is encodable with a first number of different coding modes, and

determining a rate distortion cost for a second number of coding modes, the second number being less or equal than the first number of different coding modes for the specific coding unit, the determining of the second number being such as to maximise the use of dynamically for the encoder available image frame processing resources, and

the difference between the estimated rate distortion cost for the first and the determined rate distortion cost for the second number of coding modes is the rate distortion cost error, and the difference between the rate distortion cost error and a targeted rate distortion cost error has a minimum value for a certain coding mode, said coding mode is selected to encode the coding unit.

The encoding software can be embodied in a computer program product adapted to carry out the following functions when the software is loaded onto the respective device or devices and executed on one or more processing engines such as microprocessors, FPGA, ASIC etc. :

normalizing the collection of pixels of the coding unit at level m and with index m, by dividing said coding unit with the number of pixels at the level m, and/or

for each frame, the central processing or graphics processing unit having the controller adapting the threshold of the rate distortion cost error, based on the difference between the reduced complexity and the value of the complexity target in the previous frame stored
5   in the memory.

The encoding software can be embodied in a computer program product adapted to carry out the following functions when the software is loaded onto the respective device or devices and executed on one
10   or more processing engines such as microprocessors, FPGA, ASIC etc. :

calculating the complexity for a frame in elapsed central processing or graphics processing unit processing time for encoding
15   the whole frame.

The encoding software can be embodied in a computer program product adapted to carry out the following functions when the software is loaded onto the respective device or devices and executed on one
20   or more processing engines such as microprocessors, FPGA, ASIC etc. :

treating the actual number of coding modes for per coding unit being independent of each other and/or
25

estimating the rate distortion cost comprising all coding modes by assuming said cost is proportional to a linear model of the rate distortion cost comprising less than all coding modes.

30   The encoding software can be embodied in a computer program product adapted to carry out the following functions when the software is loaded onto the respective device or devices and executed on one or more processing engines such as microprocessors, FPGA, ASIC etc. :
35

an estimated rate distortion cost is linearly proportional to the rate distortion cost comprising all coding modes multiplied by the

coefficient 0.87, 0.92, 0.94, 0.95 for the depth 0, 1, 2 and 3
respectively and/or

evaluating two groups of modes, the first group comprising the
2Nx2N merge mode and the second mode comprising all other modes.

The encoding software can be embodied in a computer program product
adapted to carry out the following functions when the software is
loaded onto the respective device or devices and executed on one
or more processing engines such as microprocessors, FPGA, ASIC etc.
:

controlling based on a root-finding method such as a bisection
method for example by executing the following steps:

$\sigma \leftarrow \sigma_0$

$equal\_sign \leftarrow True$

$function$ UPDATETHRESHOLD(C_T, C_R)

  $if(\sigma \geq 0) \neq (C_R \geq C_T)\, then$

    $\sigma \leftarrow \sigma\ \ / -2$

    $equal\_sign \leftarrow False$

  $else$

    $if\ equal\_sign\ then$

      $\sigma \leftarrow 2\sigma$

    $end\ if$

    $equal\_sign \leftarrow True$

  $end\ if$

  $E_T \leftarrow \max(0, E_T + \sigma)$

$end\ function$

Other root finding methods can be used. There are two types of
root finding methods, namely bracketed and open methods. Both
bracketed and open methods can be used with embodiments of the
present invention. Different root finding methods will have
different performance characteristics, for example bracketed
methods will converge but may take time to do so. Open methods
may be quicker but do not necessarily always have a guarantee of

31

convergence. Examples of root finding methods such as the bisection method include a false position method, the Brent Dekker method, Muller's method, Simple fixed point iteration method, Newton's method, multidimensional Newton's method, secant
5   method etc.

The encoding software can be embodied in a computer program product adapted to carry out the following functions when the software is loaded onto the respective device or devices and executed on one
10   or more processing engines such as microprocessors FPGA, ASIC etc. encoded images can be transmitted to a network for example.

The software mentioned above can be stored on a non-transitory signal storage medium, such as an optical disk (CD-ROM or DVD-
15   ROM); a magnetic tape, a magnetic disk, a ROM, or a solid state memory such as a USB flash memory or similar.

Claims

1. A method for adaptive use of image frame processing resources
   comprising a central processing unit connected to at least one
5 memory, the central processing unit being configured to receive
   frames of a video signal, the method comprising: each frame
   being split into coding units comprising pixels, defining a
   threshold for a rate distortion cost error, the difference
   between a corresponding complexity and a complexity threshold
10 is stored in the memory as a reference for the calculation of
   an updated rate distortion cost error threshold, wherein the
   difference between the complexity resulting from the updated
   rate distortion cost error threshold, and the former complexity
   threshold is smaller or equal to the difference between the
15 former complexity and the complexity threshold.

2. A method to encode video frames having coding units which
   can be coded with a first number of different coding modes,
   by, for each coding unit of the frame, the method steps of:
20 • Determining a rate distortion cost associated with encoding
   the coding unit with a coding mode for each of a chosen or
   selected second number of different coding modes, the second
   number being the same number or a reduced number of coding
   modes selected from the first number of different coding
25 modes,
   • Estimating an estimated rate distortion cost error as the
   difference between the lowest rate distortion cost amongst
   the determined rate distortion costs and an estimated lowest
   rate distortion cost amongst the first number of different
30 coding modes,
   wherein, the chosen number of the second number of different
   coding modes is selected such that the difference between the
   rate distortion cost error and a target rate distortion cost
   error is minimised.
35
3. A method of adaptive use of image frame processing resources
   for video compression, the image frame processing resources

varying dynamically in time, wherein the video signal comprises
image frames processed at a video rate, wherein an image frame
is expressed in a number of coding units, a coding unit having
a predefined maximum size, a controller being adapted to
5   receive said image frame, a coding unit being encodable with a
first number of different coding modes, the method comprising
determining a rate distortion cost for a second number of
different coding modes, the second number being less or equal
to the first number of different coding modes for the specific
10   coding unit, the difference between an estimated rate
distortion cost for the first number and the rate distortion
cost determined for the second number of coding modes is the
rate distortion cost error, and the difference between the rate
distortion cost error and a targeted rate distortion cost error
15   has a minimum value for a certain coding mode, said coding mode
is selected to encode the coding unit.

4. A method according to claim 1 or 2 or 3 further comprising
recursively splitting coding units into a quad-tree structure
20   of different coding units having levels and sub-levels, wherein
in a quad-tree structure, each level or sub-level is
recursively split into parts, a depth of a specific coding unit
being defined by the number of recursive splits of coding units
that have been used in order to reach the specific coding unit,
25   whereby the size of the specific coding unit is the maximum
size divided by the number of recursive splits.

5. A method according to claim 4 as dependent upon claim 2 or 3, a
coding unit at a certain depth being encodable with the second
30   number of different coding modes, the determining of the second
number being such as to maximise the use of dynamically for the
encoder available image frame processing resources.

6. A method according to claim 4, further comprising normalizing a
35   collection of pixels of a coding unit at level m and with index
m, by dividing said coding unit with the number of pixels at
the level m.

34

7. A method according to any of the above claims comprising calculating a complexity for a frame in elapsed central processing unit processing time for encoding the whole frame.

8. A method according to claim 7 further comprising, for each frame, adapting a threshold of the rate distortion cost error, based on the difference between a reduced complexity and a value of a complexity target in a previous frame.

9. A method according to any of the above claims, wherein the actual number of coding modes for per coding unit are treated as independent of each other.

10. A method according to any of the above claims comprising estimating the rate distortion cost comprising all coding modes by assuming said cost is proportional to a linear model of the rate distortion cost comprising less than all coding modes.

11. A method according to claim 10, wherein an estimated rate distortion cost is linearly proportional to the rate distortion cost comprising all coding modes multiplied by the coefficient 0.87, 0.92, 0.94, 0.95 for the depth 0, 1, 2 and 3 respectively.

12. A method according to any of the above claims comprising evaluating two groups of modes, the first group comprising the 2Nx2N merge mode and the second group comprising all other modes.

13. A method according to any of the above claims comprising a central processing unit implementing a controller based on a root-finding method.

14. A method according to claim 11, wherein the root-finding method is selected from a bisection method, a false position method, the Brent Dekker method, Muller's method, a Simple

fixed point iteration method, Newton's method, the
multidimensional Newton's method, and the secant method or
similar.

5   15.   A method according to any of the above claims comprising the
        following steps:
        Set initial value to the step size, assign a variable
        equal_sign to True
        If A and B can each be true or false, and A is the condition
10   that the step size is larger than or equal to zero, and B is
        the condition that the reduced complexity is larger than or
        equal to the complexity threshold, and A is different from B,
        then
        Reduce the step size to half and change direction
15   Set equal_sign to false
        If A is equal to B and equal_sign is true, then double the step
        size without changing direction,
        Set equal_sign to false, and let the rate distortion cost error
        be increased by one step size unless this is smaller than zero,
20   then let the rate distortion cost error target be zero.

    16.   A system for adaptive use of image frame processing resources
        comprising a central processing unit connected to at least one
        memory, the central processing unit being configured to receive
25   frames of a video signal, each frame being split into coding
        units comprising pixels, the central processing unit defining a
        rate distortion cost error threshold, the difference between
        the corresponding complexity and a complexity threshold is
        stored in the memory as a reference for the calculation of an
30   updated rate distortion cost error threshold, wherein the
        difference between the complexity resulting from the updated
        rate distortion cost error threshold, and the former complexity
        threshold is smaller or equal to the difference between the
        former complexity and the complexity threshold.
35

    17.   A system for adaptive use of image frame processing resources
        comprising a central processing unit connected to at least one

memory, the central processing unit being configured to receive
frames of a video signal, each frame being split into coding
units comprising pixels, a coding unit being encodable with a
first number of different coding modes, the central processing
5   unit being adapted to determine a rate distortion cost
associated with encoding the coding unit with a coding mode
for each of a chosen or selected second number of different
coding modes, the second number being the same number or a
reduced number of coding modes selected from the first
10   number of different coding modes,
And also being adapted to estimate an estimated rate
distortion cost error as the difference between the lowest
rate distortion cost amongst the determined rate distortion
costs and an estimated lowest rate distortion cost amongst
15   the first number of different coding modes,
wherein, the chosen number of the second number of different
coding modes is selected such that the difference between the
rate distortion cost error and a target rate distortion cost
error is minimised.
20

18.   A system for adaptive use of image frame processing resources
comprising a central processing unit connected to at least one
memory, the central processing unit being configured to receive
frames of a video signal, each frame being split into coding
25   units comprising pixels, a coding unit being encodable with a
first number of different coding modes, wherein the video
signal comprises the image frames processed at a video rate, a
coding unit having a predefined maximum size, the central
processing unit being adapted to determine a rate distortion
30   cost for a second number of different coding modes, the second
number being less or equal to the first number of different
coding modes for the specific coding unit, the difference
between an estimated rate distortion cost for the first number
and the rate distortion cost determined for the second number
35   of coding modes being the rate distortion cost error, and the
difference between the rate distortion cost error and a
targeted rate distortion cost error has a minimum value for a

certain coding mode, said coding mode is selected to encode the coding unit.

19. A system according to claim 18, wherein a coding unit at level m and with index m is normalised by the central processing unit dividing the coding unit with the number of pixels at the level m.

20. A system according to any of the claims 16-19, wherein for each frame, a threshold of the rate distortion cost error is adapted by the central processing unit, based on the difference between a reduced complexity and a complexity target in the previous frame.

21. A system according to claim 20, wherein the complexity for a frame is expressed in elapsed central processing unit processing time for encoding the whole frame.

22. A system according to any of the claims 16-21, wherein the central processing unit is configured to consider the actual number of coding modes for per coding unit to be independent of each other.

23. A system according to any of the claims 16-22, wherein the rate distortion cost comprising all coding modes is estimated by the central processing unit to be proportional to a linear model of the rate distortion cost comprising less than all coding modes.

24. A system according to claim 23, wherein the central processing unit is instructed to consider an estimated rate distortion cost being linearly proportional to the rate distortion cost comprising all coding modes, multiplied by the coefficient 0.87, 0.92, 0.94, 0.95 for the depth 0, 1, 2 and 3 respectively.

25.   A system according to any of the claims 16-24, wherein only
       two groups of modes are evaluated by the central processing
       unit, the first group comprising the 2Nx2N merge mode and the
       second group comprising all other modes.

26.   A system according to any of the claims 16-25, wherein the
       central processing unit implements a controller by means of a
       root-finding method.

27.   A system of claim 26, wherein the root-finding method is
       selected from a bisection method, a false position method, the
       Brent Dekker method, Muller's method, a simple fixed point
       iteration method, Newton's method, multidimensional Newton's
       method, and a secant method or similar.

28.   A controller for adaptive use of image frame processing
       resources for video compression, the image frame processing
       resources varying dynamically in time, wherein the video signal
       comprises image frames processed at a video rate, wherein an
       image frame is expressed in a number of coding units, a coding
       unit having a predefined maximum size, the controller being
       adapted to receive said image frame and for recursively
       splitting coding units into a quad-tree structure of different
       coding units having levels and sub-levels, wherein in a quad-
       tree structure, each level or sub-level is recursively split
       into parts, a depth of a specific coding unit being defined by
       the number of recursive splits of coding units that have been
       used in order to reach the specific coding unit, whereby the
       size of the specific coding unit is the maximum size divided by
       the number of recursive splits, a coding unit at a certain
       depth being encodable with a first number of different coding
       modes, further comprising the controller bring adapted to
       determine a rate distortion cost for a second number of coding
       modes, the second number being less or equal to the first
       number of different coding modes for the specific coding unit,
       the controller being adapted for determining of the second

number such as to maximise the use of dynamically for the
encoder available image frame processing resources,
the difference between the rate distortion cost for the first
and second number of coding modes is the rate distortion cost
5   error, and the difference between the rate distortion cost
error and a targeted rate distortion cost error has a minimum
value for a certain coding mode, the controller being adapted
to select said certain coding mode to encode the coding unit.

10   29. A controller according to claim 28 being further adapted for
normalizing the collection of pixels of the coding unit at
level m and with index m, by dividing said coding unit with the
number of pixels at the level m.

15   30. A controller according to claim 28 or 29, wherein the
controller is adapted, for each frame, to adapt the threshold
of the rate distortion cost error, based on the difference
between the reduced complexity and the value of the complexity
target in the previous frame stored in the memory.
20
     31. A controller according to any of the claims 28 to 30, wherein
the controller is adapted to calculate the complexity for a
frame in elapsed central processing unit processing time for
encoding the whole frame.
25
     32. A controller according to any of the claims 28 to 31, wherein
the controller is adapted to treat the actual number of coding
modes for per coding unit as being independent of each other.

30   33. A controller according to any of the claims 28 to 32, wherein
the controller is adapted to estimate the rate distortion cost
comprising all coding modes by assuming said cost is
proportional to a linear model of the rate distortion cost
comprising less than all coding modes.
35
     34. A controller according to claim 33, wherein an estimated rate
distortion cost is linearly proportional to the rate distortion

cost comprising all coding modes multiplied by the coefficient
0.87, 0.92, 0.94, 0.95 for the depth 0, 1, 2 and 3
respectively.

35.   A controller according to any of the claims 28 to 34 the
controller being adapted to evaluate two groups of modes, the
first group comprising the 2Nx2N merge mode and the second
group comprising all other modes.

36.   A controller according to any of the claims 28 to 35, being
adapted for implementing the controller based on a root-finding
method.

37.   A controller according to claim 36, further adapted to
execute the following steps:
Set initial value to the step size, assign a variable
equal_sign to True If A and B can each be true or false, and A
is the condition that the step size is larger than or equal to
zero, and B is the condition that the reduced complexity is
larger than or equal to the complexity threshold, and A is
different from B, then Reduce the step size to half and change
direction Set equal_sign to false If A is equal to B and
equal_sign is true, then double the step size without changing
direction, Set equal_sign to false, and let the rate distortion
cost error be increased by one step size unless this is smaller
than zero, then let the rate distortion cost error target be
zero.

38.   A Controller for adaptive use of image frame processing
resources, the controller comprising a central processing unit
connected to at least one memory, the central processing unit
being configured to receive frames of a video signal, each
frame being split into coding units comprising pixels, the
central processing unit defining a rate distortion cost error
threshold, the difference between the corresponding complexity
and a complexity threshold is stored in the memory as a
reference for the calculation of an updated rate distortion

cost error threshold, wherein the difference between the
complexity resulting from the updated rate distortion cost
error threshold, and the former complexity threshold is smaller
or equal to the difference between the former complexity and
the complexity threshold.

39.   A controller according to claim 38, wherein a coding unit at
level m and with index m is normalised by instructing the
central processing unit dividing the coding unit with the
number of pixels at the level m.

40.   A controller according to claim 38 or 39, wherein for each
frame, the threshold of the rate distortion cost error being
adapted by the central processing unit, based on the difference
between the reduced complexity and the complexity target in the
previous frame.

41.   A controller according to any of claims 38-40, wherein the
complexity for a frame is expressed in elapsed central
processing unit processing time for encoding the whole frame.

42.   A controller according to any of claims 38-41 wherein the
central processing unit is being instructed to consider the
actual number of coding modes for per coding unit being
independent of each other.

43.   A controller according to any of the claims 38 to 42, wherein
the rate distortion cost comprising all coding modes is
estimated by the central processing unit to be proportional to
a linear model of the rate distortion cost comprising less than
all coding modes.

44.   A controller according to claim 43, wherein the central
processing unit is instructed to consider an estimated rate
distortion cost being linearly proportional to the rate
distortion cost comprising all coding modes, multiplied by the

42

coefficient 0.87, 0.92, 0.94, 0.95 for the depth 0, 1, 2 and 3 respectively.

45.   A controller according to any of claims 38 to 44, wherein only two groups of modes are evaluated by the central processing unit, the first group comprising the 2Nx2N merge mode and the second mode comprising all other modes.

46.   A method according to any of the claims 1 to 15, further comprising transmitting an encoded frame or image over a network.

47.   A system according to any of the claims 16 to 27, adapted for transmitting an encoded frame or image over a network.

48.   A controller according to any of the claims 28 to 45, adapted for transmitting an encoded frame or image over a network.

49.   A computer program product comprising code that when executed on a processing engine executes any of the method steps of claims 1 to 15.

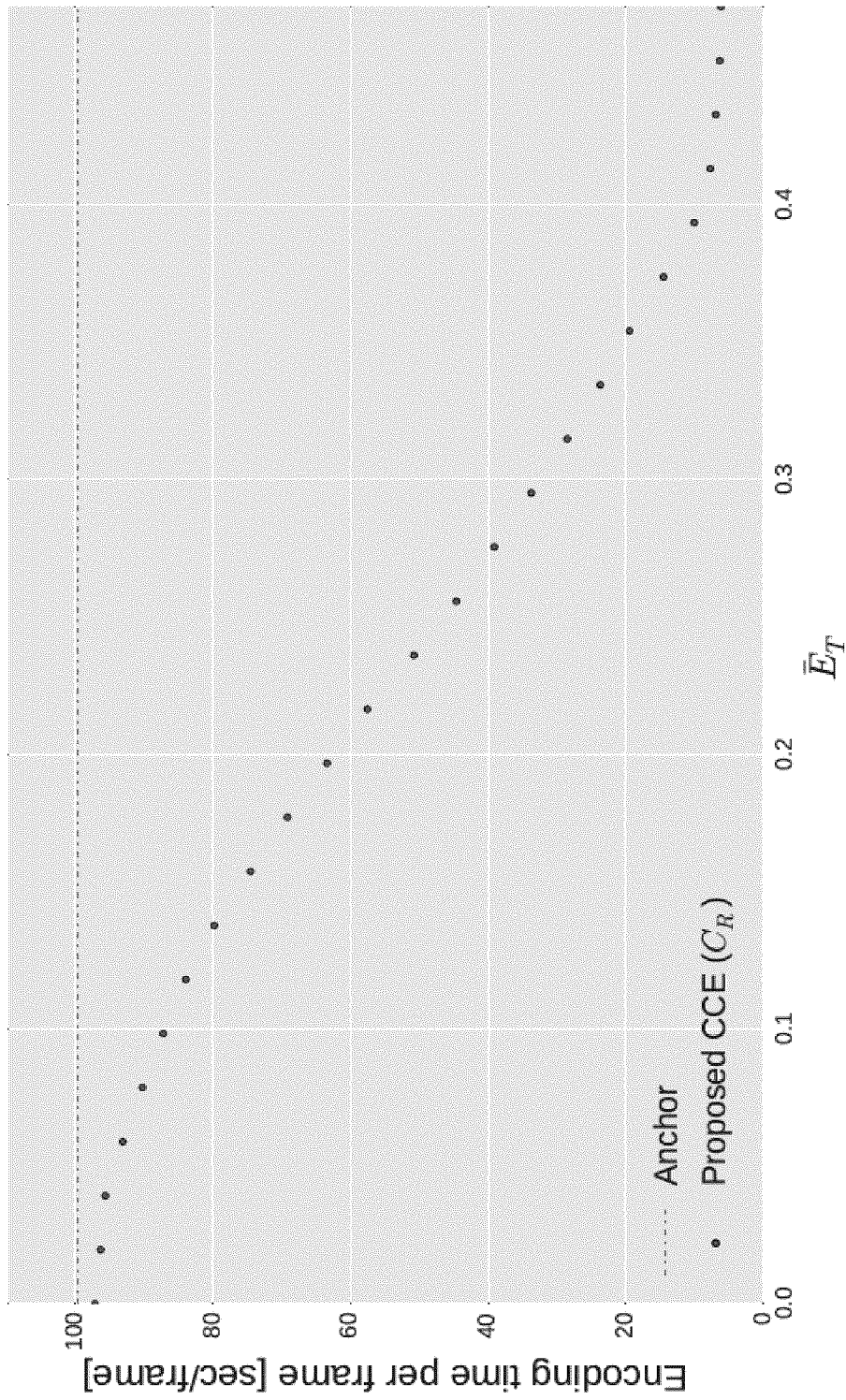50.   A non-transitory signal storage storing the computer program product of claim 49.

Figure 1

Figure 2

$$\sigma \leftarrow \sigma_0$$

$$equal\_sign \leftarrow True$$

$$function\ \text{UPDATETHRESHOLD}(C_T, C_R)$$

$$\quad if\ (\sigma \geq 0) \neq (C_R \geq C_T)\ then$$

$$\quad\quad \sigma \leftarrow \sigma\ /-2$$

$$\quad\quad equal\_sign \leftarrow False$$

$$\quad else$$

$$\quad\quad if\ equal\_sign\ then$$

$$\quad\quad\quad \sigma \leftarrow 2\sigma$$

$$\quad\quad end\ if$$

$$\quad\quad equal\_sign \leftarrow True$$

$$\quad end\ if$$

$$\quad E_T \leftarrow \max(0, E_T + \sigma)$$

$$end\ function$$

Figure 3

Figure 4a)

5/11
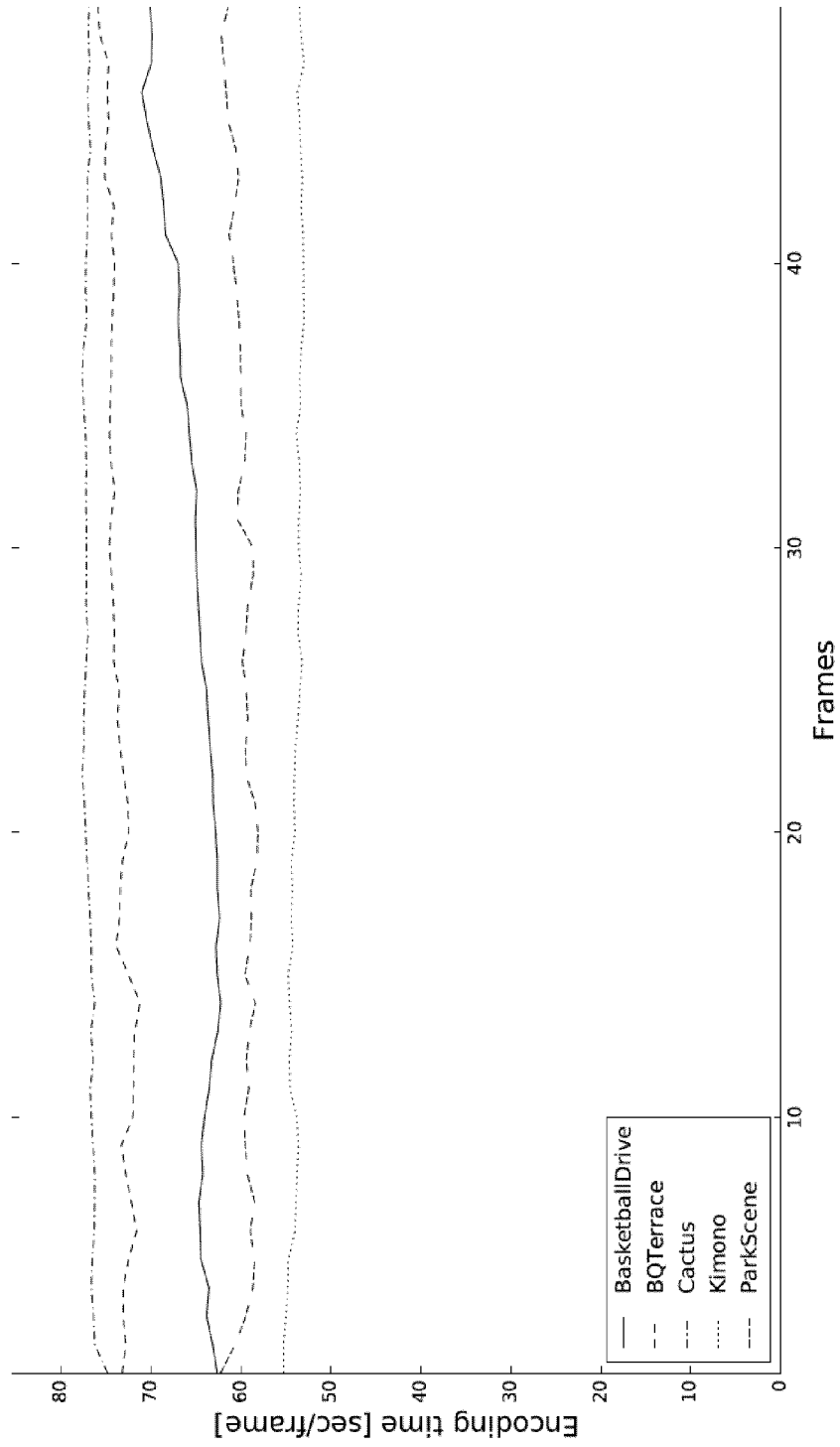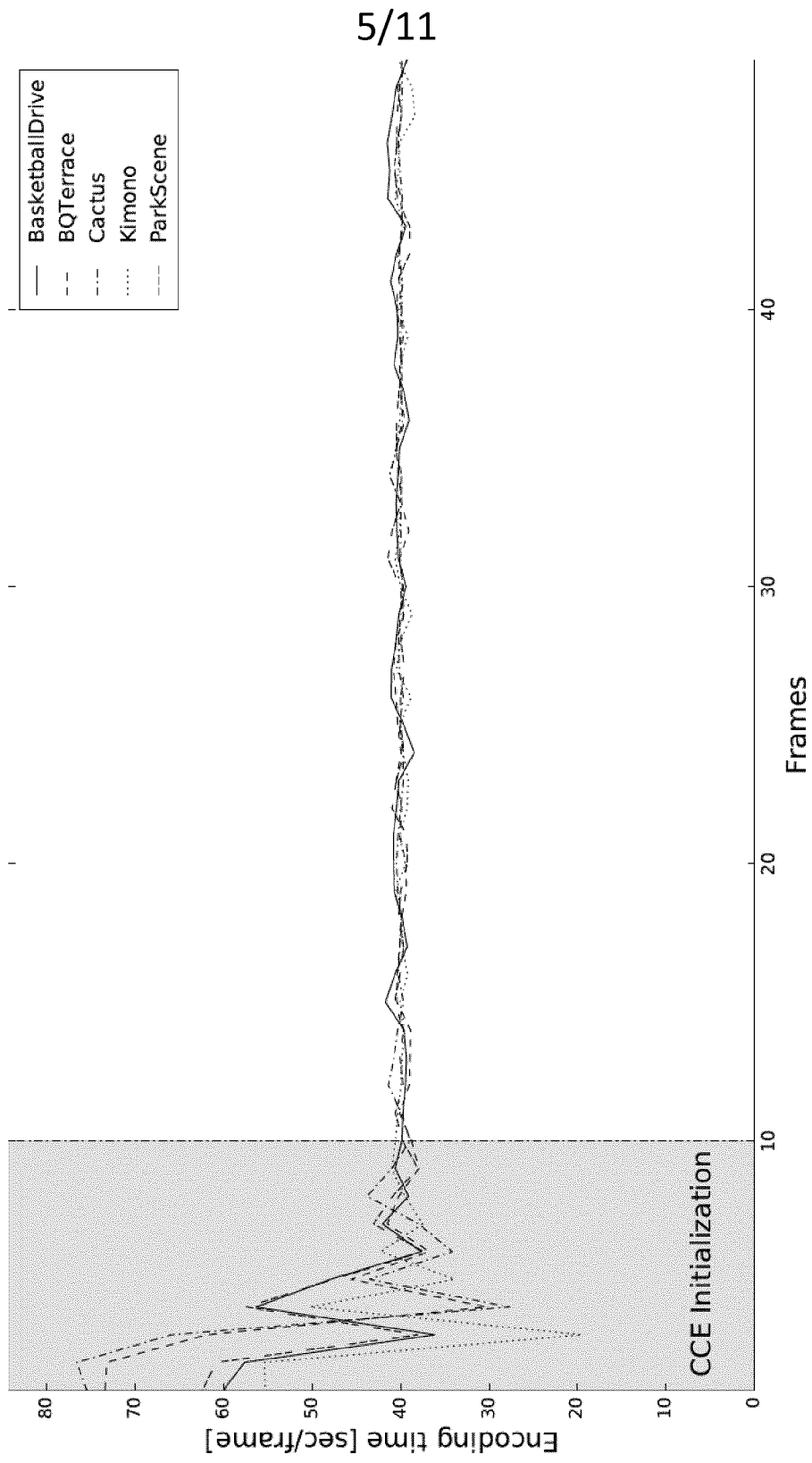
Figure 4b)

Sequence	Anchor			CCE		
	ET (sec)	PSNR (dB)	Rate (Mbit/s)	ET (sec)	PSNR (dB)	Rate (Mbit/s)
basketballdrive	65.56	42.69	55.04	40.26	42.59	53.13
bqterrace	73.89	42.93	135.55	39.95	42.62	137.52
cactus	77.09	41.70	92.99	40.10	41.57	94.52
kimono	53.71	43.63	12.69	39.87	43.59	12.49
parkscene	59.82	42.26	20.99	40.09	42.18	20.88

(c)

Figure 4c)
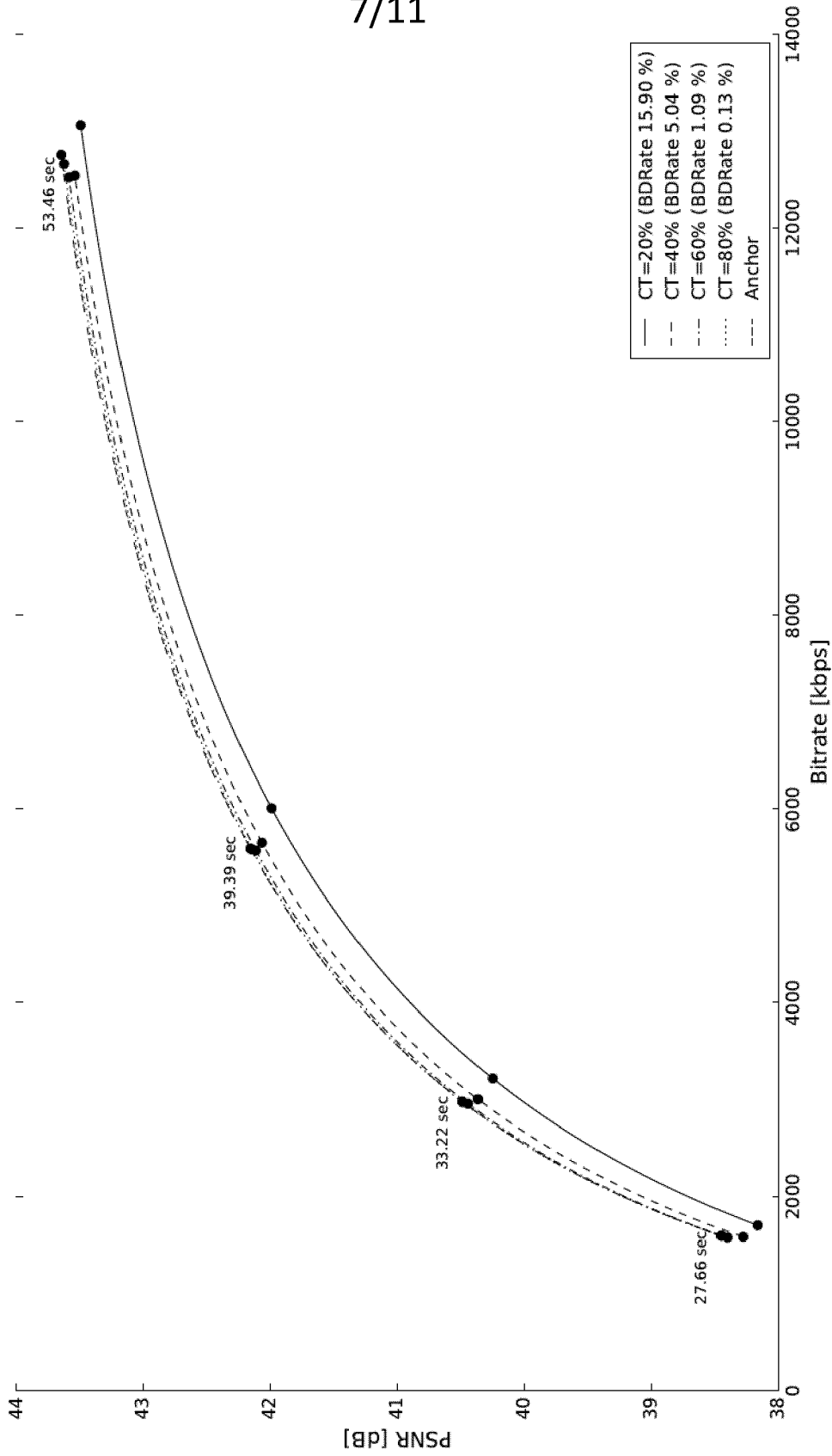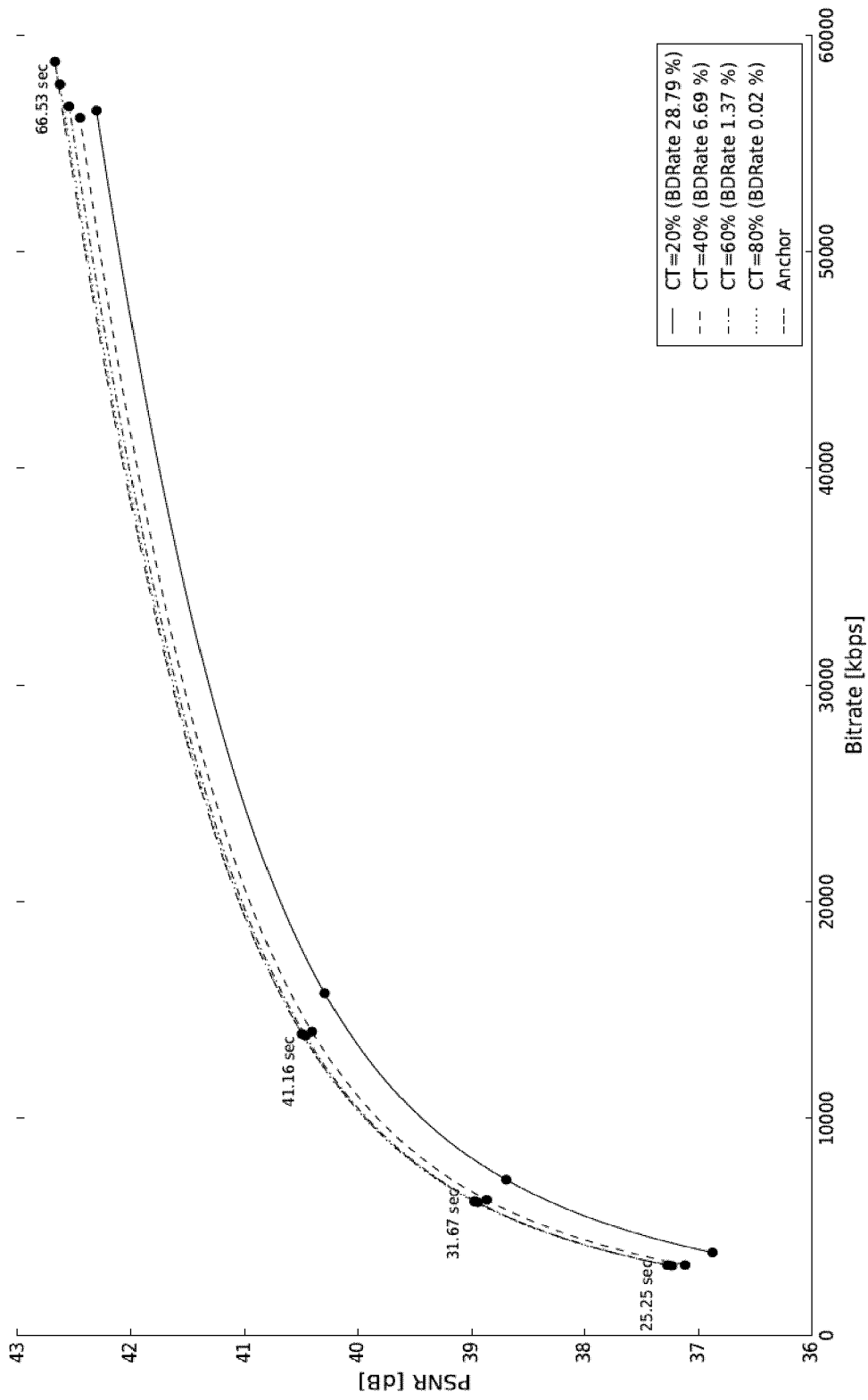
Figure 5a)

Figure 5b)

Sequence	Proposed Method								Fixed Complexity Reduction			
	80%		60%		40%		20%		ESD[5]		MD2	
	ET	BDR	ET	BDR	ET	BDR	ET	BDR	ET	BDR	ET	BDR
basketballdrive	0.79	0.07	0.59	1.34	0.40	6.47	0.20	28.23	0.74	2.22	0.42	8.98
bqterrace	0.79	1.72	0.59	5.48	0.39	11.17	0.20	24.33	0.77	1.56	0.43	6.90
cactus	0.78	0.18	0.59	2.83	0.39	10.49	0.20	28.92	0.76	2.00	0.41	12.20
kimono	0.79	0.11	0.59	1.08	0.40	4.98	0.20	15.93	0.78	2.37	0.40	3.24
parkscene	0.79	0.00	0.59	3.45	0.40	13.52	0.20	35.55	0.69	2.28	0.43	14.18

(c)

Figure 5c)
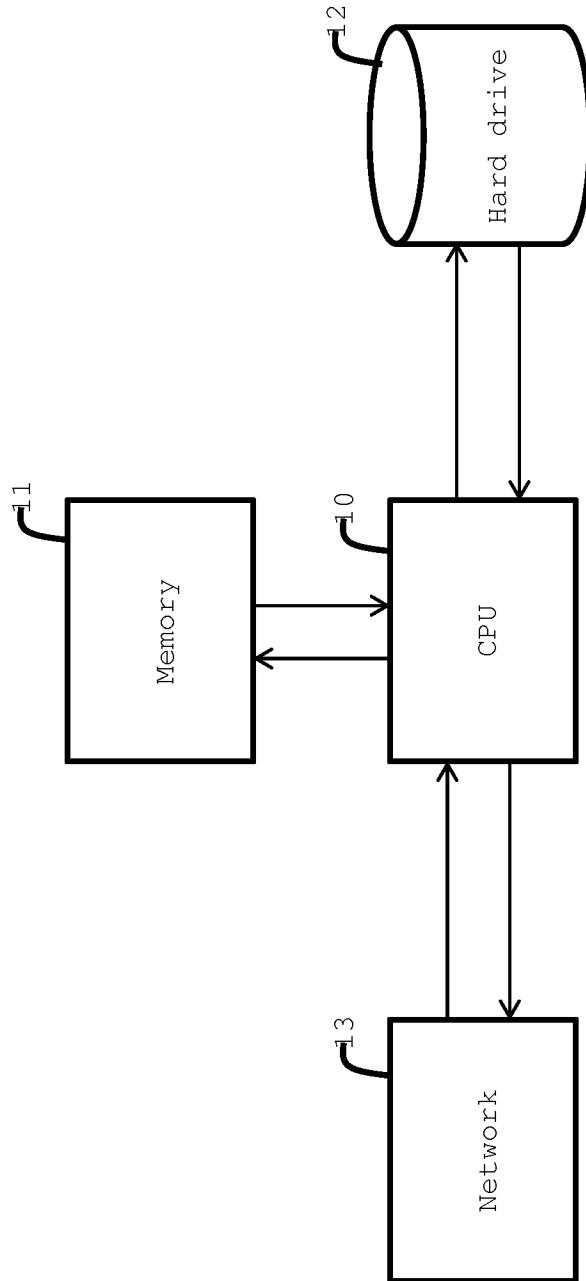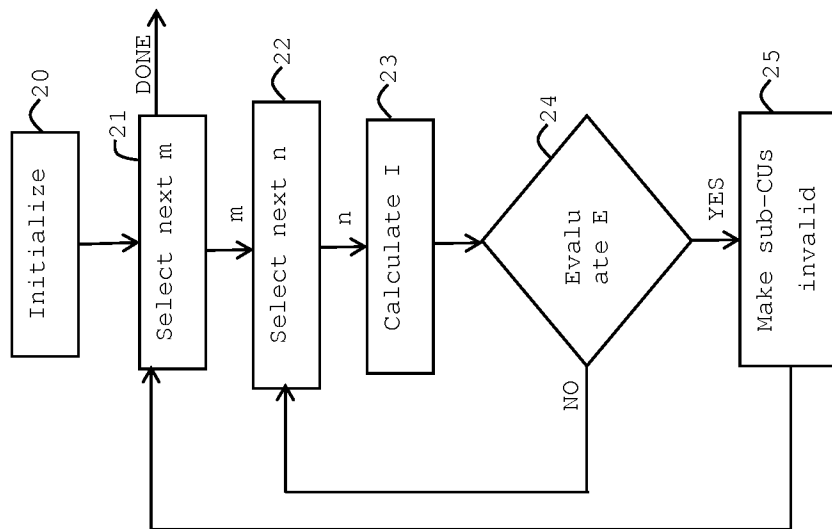
Figure 6

Figure 7

INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N19/147 H04N19/103 H04N19/109 H04N19/127 H04N19/11
H04N19/19

ADD.

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)
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 practicable, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	LAROCHE G ET AL: "Encoder speed-up for the MV cost estimation", 8. JCT-VC MEETING; 99. MPEG MEETING; 1-2-2012 - 10-2-2012; SAN JOSE; (JOINT COLLABORATIVE TEAM ON VIDEO CODING OF ISO/IEC JTC1/SC29/WG11 AND ITU-T SG.16); URL: HTTP://WFTP3.ITU.INT/AV-ARCH/JCTVC-SITE/,, no. JCTVC-H0178, 20 January 2012 (2012-01-20), XP030111205, paragraph [0001]; figures 1,2 ----- -/--	1-50

| X | Further documents are listed in the continuation of Box C. | | X | 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 application or patent 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
1 June 2017	14/06/2017

Name and mailing address of the ISA/ European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Authorized officer Kuhn, Peter

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

C(Continuation).   DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	VERMEIR THIJS ET AL: "Real-time complexity constrained encoding", 2016 IEEE INTERNATIONAL CONFERENCE ON IMAGE PROCESSING (ICIP), IEEE, 25 September 2016 (2016-09-25), pages 819-823, XP033016639, DOI: 10.1109/ICIP.2016.7532471 [retrieved on 2016-08-03] paragraph [0002] - paragraph [0003] -----	1-50
X,P	VERMEIR THIJS ET AL: "Low Delay Complexity Constrained Encoding", 2016 DATA COMPRESSION CONFERENCE (DCC), IEEE, 30 March 2016 (2016-03-30), page 635, XP033027795, DOI: 10.1109/DCC.2016.54 [retrieved on 2016-12-15] the whole document -----	1-50
A	US 2008/112481 A1 (HSAING SHIH-TA [US] ET AL) 15 May 2008 (2008-05-15) paragraph [0035] - paragraph [0064]; figures 4-8 -----	1-50
A	TIESONG ZHAO ET AL: "Flexible Mode Selection and Complexity Allocation in High Efficiency Video Coding", IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING, vol. 7, no. 6, 1 December 2013 (2013-12-01), pages 1135-1144, XP55376395, US ISSN: 1932-4553, DOI: 10.1109/JSTSP.2013.2271421 cited in the application paragraph [00II] - paragraph [00IV] -----	1-50
A	CORREA GUILHERME ET AL: "Complexity scalability for real-time HEVC encoders", JOURNAL OF REAL-TIME IMAGE PROCESSING, SPRINGER, DE, vol. 12, no. 1, 5 January 2014 (2014-01-05), pages 107-122, XP035942313, ISSN: 1861-8200, DOI: 10.1007/S11554-013-0392-8 [retrieved on 2014-01-05] cited in the application paragraph [0001] - paragraph [0005] -----	1-50

-/--

1

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	XIANG LI ET AL: "Rate-Complexity-Distortion Optimization for Hybrid Video Coding", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE SERVICE CENTER, PISCATAWAY, NJ, US, vol. 21, no. 7, 1 July 2011 (2011-07-01), pages 957-970, XP011358100, ISSN: 1051-8215, DOI: 10.1109/TCSVT.2011.2133750 the whole document -----	1-50

1

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2008112481 A1	15-05-2008	NONE	